



**Universidad**  
Zaragoza

## Trabajo Fin de Grado

Caracterización térmica de una plataforma  
multiprocesador refrigerada por aire

Thermal characterization of an air-cooled  
multiprocessor

Autor/es

Sergio Escolar de Torres-Solanot

Director/es

Montijano Muñoz, Eduardo  
Suárez Gracia, Darío

Escuela de Ingeniería y arquitectura  
2019

Sergio Escolar de Torres-Solanot: *CARACTERIZACIÓN TÉRMICA DE UNA PLATAFORMA  
MULTIPROCESADOR REFRIGERADA POR AIRE*

Gracias a Eduardo y Darío

Me gustaría expresar mi más profundo agradecimiento, por hacer posible la realización de este trabajo, transmitiéndome cada día energía, positividad y motivación en vuestras respectivas áreas

Gracias a mis padres, a mis hermanos y a toda mi familia

Porque han sido parte fundamental para ser quien soy y hacia quienes sólo puedo expresar mi más sincero agradecimiento por apoyarme en todo momento

Y gracias en especial a María José, porque sé que hay que tener mucha paciencia para lidiar con un ingeniero



# RESUMEN

## CARACTERIZACIÓN TÉRMICA DE UNA PLATAFORMA MULTIPROCESADOR REFRIGERADA POR AIRE

En la actualidad la informática y los dispositivos electrónicos se están convirtiendo cada vez más en parte indispensable de nuestra vida. Desde que aparecieron los primeros ordenadores en los años 40, han evolucionado de forma exponencial, implantando y adaptando su uso en cada vez más ámbitos de la sociedad. El problema es que cada vez los dispositivos son tan pequeños y potentes que generan mucho calor, causando unas grandes dificultades al tratar de disipar toda esa energía generada.

Las soluciones que existen para remediar el problema se pueden dividir en dos, por un lado, soluciones hardware y, por otro lado, soluciones software. Sin embargo, en el estado del arte no se suele tener en cuenta la temperatura externa, siendo que cuando la temperatura del entorno aumenta, la transferencia del calor desde los dispositivos al exterior se vuelve más lenta, incluso puede llegar a suceder que no exista, haciendo que los dispositivos trabajen a temperaturas más elevadas de las recomendadas por los fabricantes.

En este trabajo se pretende caracterizar la influencia que tiene la temperatura exterior sobre una placa a partir de datos experimentales con herramientas como Matlab/Simulink, dicho ajuste se pretende realizar mediante modelos simplificados estudiados en el estado del arte. Además, debido al alto coste de las cámaras térmicas, se diseña una estructura en la que se puede disponer de un mayor control que en una estructura abierta, estresando térmicamente los *cores* de la placa mediante variaciones en la temperatura externa y refrigeración por aire. El objetivo es obtener modelos que puedan establecer unas bases para el diseño de controles que minimicen sus efectos.

Este proyecto aúna conocimientos informáticos, termodinámicos y de ingeniería de control. Para la realización de los experimentos, se utiliza una placa de desarrollo con un multiprocesador sin disipador empleado en móviles y otros dispositivos como routers.



# ÍNDICE GENERAL

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>1</b>
1.1	Motivación.....	1
1.2	Objetivos del proyecto .....	2
1.3	Alcance .....	3
1.4	Organización .....	4
<b>2</b>	<b>ESTADO DEL ARTE .....</b>	<b>5</b>
2.1	Estado del arte .....	5
2.2	Plataforma multiprocesador .....	6
2.3	Generación de calor .....	7
2.4	Fenómenos de transferencia de calor .....	8
2.4.1	Conducción .....	8
2.4.2	Convección .....	9
2.4.3	Radiación.....	11
2.5	Disipación de calor .....	11
2.6	Gestión en Linux de la temperatura y frecuencia .....	13
2.6.1	Gestión de temperatura .....	13
2.6.2	Gestión de frecuencia .....	14
2.7	Identificación del sistema térmico .....	15
2.8	Identificación de los parámetros del sistema.....	16

2.8.1	Identificación de una respuesta a escalón .....	17
<b>3</b>	<b>PLATAFORMA DE CONTROL TÉRMICO DE BAJO COSTE.....</b>	<b>19</b>
3.1	Requerimientos.....	19
3.2	Diseño físico.....	20
3.2.1	Instrumentación refrigeración y temperatura .....	21
3.2.2	Evolución de la estructura de pruebas.....	22
3.2.3	Sistema de medición externo .....	24
3.3	Adquisición de datos térmicos .....	27
3.3.1	Medida interna .....	27
3.3.2	Medidas externas .....	28
3.3.3	Sincronización en Matlab.....	28
3.4	Ejemplo de uso.....	32
<b>4</b>	<b>CARACTERIZACIÓN TÉRMICA .....</b>	<b>34</b>
4.1	Diseño de los experimentos.....	34
4.2	Resultados obtenidos .....	35
4.2.1	Influencia estructura abierta-cerrada .....	37
4.2.2	Influencia de la temperatura del entorno-Caudal aire.....	38
4.2.3	Control con estrés de la placa .....	41
4.3	Análisis de resultados con Matlab y determinación de parámetros .....	41
4.4	Lecciones aprendidas.....	45
<b>5</b>	<b>CONCLUSIONES .....</b>	<b>46</b>



- BIBLIOGRAFÍA .....	48
A. Plataforma y obtención de datos .....	52
A.1 Evolución de la estructura .....	52
A.2 Sistema de medición del sensor del procesador .....	54
A.3 Código desarrollado para la obtención de los datos del procesador .....	55
A.3.1 Ejecutable fijar frecuencia .....	55
A.3.2 Ejecutable leer temperatura y frecuencia .....	55
A.3.3 Ejecutable PID .....	56
B. CONTROLADOR PID .....	57
B.1 Concepto y funcionamiento de un controlador PID .....	57
C. CÓDIGO Y RESULTADOS EN MATLAB .....	59
C.1 Código en Matlab .....	59
C.1.1 Tratamiento temperatura-frecuencia .....	59
C.1.2 Tratamiento temperatura interna procesador .....	59
C.1.3 Tratamiento temperaturas externas .....	60
C.1.4 Tratamiento sincronización de medidas .....	61
C.2 Pruebas realizadas para la obtención de los parámetros del sistema .....	63
C.3 Pruebas para la comparación entre el control por defecto y el PID .....	67
C.4 Gráficas de ajustes obtenidos .....	69



# 1 INTRODUCCIÓN

---

## 1.1 Motivación

Una de las mayores preocupaciones durante los últimos años en el diseño de procesadores es la generación de calor en el procesador y su disipación. Los dispositivos móviles mejoran su diseño a un ritmo vertiginoso, de hecho, un dispositivo móvil tiene más potencia que un superordenador de tan solo hace unos años. Pero no solo este tipo de dispositivos, sino también sistemas como coches autónomos, robots, monitores médicos o sistemas de vigilancia han visto incrementada su capacidad de cálculo permitiéndoles ejecutar elevadas cargas.

El incremento de la demanda de cálculo lleva consigo un incremento en la cantidad de calor a disipar. La temperatura que alcanza un sistema *ciberfísico* es difícil de analizar en cuanto a cómo se ha producido y que código y elementos producen un mayor calentamiento (cada procesador tiene su propia microarquitectura) y, además, en cuanto a su medición se ha de conocer información sobre el procesador que rara vez es pública. En concreto en el mundo de la telefonía, aunque también en multitud de sistemas *ciberfísicos*, los procesadores no pueden alcanzar temperaturas muy elevadas ya que podrían afectar al chip o a la batería, llegando a causar una reducción del rendimiento [1], un daño permanente e incluso llegando a explotar o a quemar en las manos del usuario [2]. Por otro lado, la limitación del espacio en este tipo de dispositivos imposibilita el uso de técnicas mecánicas como la refrigeración mediante ventiladores o refrigeradores líquidos, como se implementa en otros ordenadores. Por lo tanto, la importancia del calor en el procesador y su consumo se ha convertido en uno de los principales cuellos de botella que limita el rendimiento de este tipo de dispositivos [3].

Hoy en día, todos los dispositivos electrónicos tienen un control de temperatura, estos pueden diferir mucho entre ellos, pero ninguno tiene en cuenta el entorno y como afecta al comportamiento del dispositivo. Los cambios bruscos de temperatura pueden provocar pérdidas de datos en los equipos informáticos, ya que puede haber fallos en los discos duros. Un claro caso lo vivimos a diario, ya que la comodidad de usar un dispositivo móvil en contacto directo con nuestro cuerpo, el cual se encuentra aproximadamente a 36 °C, dificulta la transferencia de calor provocando un recalentamiento en el interior de los dispositivos electrónicos, o los ordenadores que deberían trabajar a una temperatura exterior de 22 °C para un buen funcionamiento, pero cuando se superan

los 30 °C se incrementan los peligros para nuestros ordenadores [4]. Por ese motivo en verano suceden el doble de averías.

Por ello, en este trabajo se propone una solución la cual se va a centrar en ajustar la influencia que produce la variación de temperatura externa en la placa multiprocesador, estresando térmicamente los *cores* haciendo que corran dentro de *Linux*. Caracterizar dicha perturbación térmica puede permitir determinar de forma más exacta la temperatura que se alcanza con cada instrucción y cualquier ejecución de un programa, lo cual establecería unas bases para el diseño de mejores controles de gestión de la temperatura. Dicha caracterización puede ser útil también para sistemas de control de temperatura pasivos sin necesidad de usar dispositivos externos al procesador, es decir, técnicas software.

Esto supondría la reducción de costes en la refrigeración y el aumento de la eficiencia del dispositivo. Por otro lado, en este ámbito se encuentran muchas dificultades por la baja existencia de trabajos previos que realicen mediciones de temperatura en un hardware real.

## 1.2 Objetivos del proyecto

El objetivo principal del trabajo es la caracterización térmica de la perturbación que supone la temperatura exterior en una plataforma multiprocesador refrigerada por aire. Este objetivo deriva en tener que alcanzar los siguientes subobjetivos:

1. Entender cómo se genera el calor en una plataforma multiprocesador y como se disipa desde un punto de vista termodinámico, y cuál es la gestión que realiza *Linux* en la placa. Lo que ayudará a entender cuáles son las variables que entran en juego en el calentamiento y enfriamiento de un procesador.
2. Adquisición de conocimientos en la rama automática e informática y nociones en la adquisición y tratamiento de datos.
3. Diseñar una plataforma de control térmico de bajo coste en el que se pueda someter la placa a un estrés térmico controlado, donde se tome tanto la temperatura del sensor del procesador como la temperatura externa, y la sincronización de estas medidas.
4. Diseño de los experimentos necesarios para la obtención de los modelos y entendimiento de las variables que entran en juego.
5. Ajuste de modelos a los datos empíricos, comparativa de los resultados y análisis de los resultados. En los que se incluye además un análisis empírico del control PID [6] frente al control por defecto, cuando se somete a un estrés térmico.

### 1.3 Alcance

El presente trabajo engloba conocimientos de control y sistemas automáticos, termodinámicos, e informáticos.

Durante la realización de este trabajo se han realizado las siguientes tareas:

- Estudio del trabajo previo realizado por Pablo Hernández, motivado por la utilización de la misma placa para los experimentos en su trabajo de fin de carrera [6] trabajo en el que se diseñó un *Governor* con un control PID automático en dicha plataforma.
- Familiarizarse con la plataforma y la interfaz de la placa; realizar pruebas en lazo abierto y lazo cerrado según los parámetros del control PID introducidos (proporcional, integral y derivativo).
- Por un lado, debido a la parte de carácter informático del proyecto, se ha necesitado adquirir nociones de programación y adquisición de datos.
- Por otro lado, en la rama automática se tenía conocimientos previos, pero son conceptos en los que se tuvo que invertir tiempo en profundizar, debido a la metodología que se pretende emplear en este trabajo.
- Diseño de una plataforma de control de bajo coste en la que se pueda estresar térmicamente la placa multiprocesador y elección de la instrumentación a emplear.
- Análisis de los puntos en los que puede interesar medir la temperatura, y comprobación del correcto funcionamiento de la plataforma de medición, tomando como referencia la diseñada por Sergio Gutierrez en su trabajo de fin de carrera [7], que diseña una plataforma de medida en la que se toma tanto la del sensor del procesador, como la del sistema externo autónomo.
- Desarrollo del código en Matlab que transforma los datos recogidos por el sistema de medición externo y los datos del sensor interno del procesador. Realizándose la posterior sincronización para su posterior representación óptima.
- Estudio de las pruebas experimentales a realizar que puedan ser de interés tanto en la caracterización del sistema como en la de su comprensión.
- Análisis, tratamiento y estudio de todos los datos obtenidos en las pruebas, obteniendo las conclusiones y valoraciones del proyecto con ayuda de las herramientas *Matlab* y *Simulink*. Además, se puso a prueba la placa realizando un experimento con un control PID implementado en dicha placa, comparándolo con la eficacia del control por defecto.

En la Ilustración 1.1 se muestra el diagrama de Gantt en el que se puede observar la distribución de esfuerzos realizados en las distintas partes.

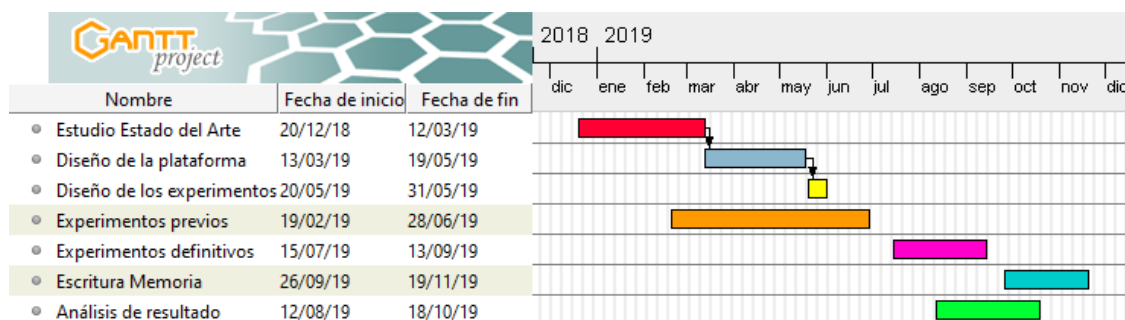


Ilustración 1.1 Diagrama de Gantt.

## 1.4 Organización

El documento se encuentra dividido en 5 capítulos. Tras la introducción en el [capítulo 2](#) se habla del estado del arte actual sobre los fundamentos teóricos que existen tras la generación, transferencia y disipación de calor en una placa, literatura de sistemas actuales, gestión en *Linux*, simplificaciones adoptadas en los modelos y, por último, herramientas empleadas en la identificación de los parámetros. En el [capítulo 3](#), se explica el desarrollo de una plataforma de control térmico de bajo coste, los requerimientos necesarios y elaboración del código informático para el tratamiento e implementación de una correcta adquisición de datos. Además, se describe el funcionamiento en un ejemplo de uso recogiendo el código en anexos, para que cualquier persona pueda hacer uso de ello. En el [capítulo 4](#) se realiza tanto el diseño de experimentos como el análisis de resultados. El último [capítulo 5](#) concluye y comenta las posibles líneas de futuro.

# 2 ESTADO DEL ARTE

---

En este capítulo se realiza un estudio preliminar sobre las soluciones actuales de gestión de la temperatura, la generación y disipación de calor de la placa al exterior, cómo se gestiona en Linux y de qué forma se identificarán los modelos a partir de datos experimentales.

## 2.1 Estado del arte

Durante los últimos años, el control de temperatura ha suscitado un gran interés, exponiendo una gran cantidad de propuestas para dicho control. Entre ellas, se pueden clasificar en dos: Soluciones basadas en hardware y soluciones software pasivas.

Las técnicas hardware mediante el uso de elementos de los dispositivos como Bubbico et al. [8] que usa la batería para disipar el exceso de calor. Lou [10], Sabbah et al. [11] y Yang et al. [12] son solo algunos de los que utilizan diferentes métodos de enfriamiento de baterías por aire o el en caso de Valøen et al. [9] y Yueh [13] mediante refrigeración proponiendo un sistema de refrigeración líquida con el fin de disminuir el consumo, incluso propuestas a partir de refrigeración por tubería de calor Gou et al. [15] o con materiales que cambian de fase (Bubbico et al. [16]; Ling et al. [14]). Sin embargo, aunque estas técnicas mitigan el problema de disipación, requieren un coste económico y medioambiental.

En cuanto a las técnicas software pasivas, se enfocan más en dispositivos *ciberfísicos*. Estas soluciones pueden diferenciarse entre las que usan herramientas de ingeniería de control y las basadas en heurísticas, estas últimas suelen enfocarse en el consumo energético, como Park [17] que llega a parar las tareas que se ejecutan de fondo, priorizando las tareas que se realizan en primer plano, con el inconveniente de que podamos estar influyendo en tareas que también sean importantes. Por otro lado, en las basadas en herramientas de control se han optado por diferentes propuestas como la del uso de un controlador MIMO (*multiple-input multiple-output*), que busca la mejor relación rendimiento-consumo propuesta por Pothukuchi et al. [18], dicha propuesta no ha sido evaluada en una plataforma real, o la propuesta de Maggio et al. [19] el cual propone un control PID, con el objetivo de mantener un rendimiento constante, asegurando calidad de servicio sin considerar la temperatura. Por último, nombrar el control IPA diseñado por la empresa ARM [5] que propone un control el cual restringe las frecuencias disponibles a partir de modelos de potencia.

Estas propuestas y soluciones son de gran interés en este ámbito, pero en ellos no se tiene en cuenta la temperatura externa, es por ello por lo que este trabajo se centrará en ajustar la perturbación que produce la variación de la temperatura externa en la placa multiprocesador, con el objetivo de minimizar los efectos pudiendo llegar a mitigar su influencia a partir del término integral del control PID. De la misma forma, se pueden introducir términos en la acción que se denomina ‘*Prealimentación de perturbaciones*’ [20], cuyo objetivo es la eliminación de las perturbaciones. Para predecir adecuadamente cualquier sistema, el modelo matemático es de gran importancia, pero en la mayoría de los casos requiere mucho tiempo en términos de tiempo de configuración y cálculo. En este trabajo se va a partir de datos experimentales ajustándolos a modelos razonablemente simplificados, con el objetivo de comprobar la viabilidad de las suposiciones. Además, trabajando con la temperatura máxima medida del procesador se evita la necesidad del conocimiento de modelos de energía complejos que describan el sistema.

## 2.2 Plataforma multiprocesador

En este proyecto se va a utilizar la plataforma de desarrollo *ICFC6410* de la empresa *Inforce*, la cual es un ordenador de placa reducida o también conocida como SBC (*Single Board Computer*). Dispone de un procesador *Qualcomm Snapdragon 600* con cuatro núcleos krait 300 hasta 1.7 GHz de frecuencia de reloj. Dicha placa viene sin disipador interno y únicamente dispone de un empaquetado.

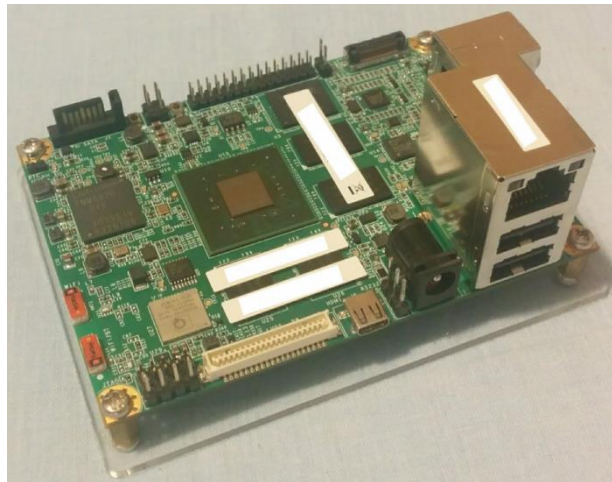


Ilustración 2.1 Imagen de la plataforma multiprocesador



## 2.3 Generación de calor

La energía en un procesador es usada tanto para la transmisión de información a través del cableado como para la conmutación de los transistores.

Un procesador es un circuito eléctrico y como en todo circuito eléctrico hay parte de la energía que no se consume haciendo operaciones por el sistema, sino que se transforma en calor por el rozamiento de los electrones en las paredes de los conductores, este se traduce lógicamente en un aumento del consumo total, puesto que hay que suministrar una mayor potencia de la que se requeriría [24].

Por un lado, la corriente eléctrica se transforma en calor mediante el consumo estático, el cual es el consumo de los transistores que no se emplea en la conmutación y se debe a fugas. Dicho consumo se produce cuando la tensión aplicada no alcanza el punto de activación la cual se debe a la corriente subumbral. El transistor está apagado teóricamente, pero la realidad es que por motivos de fabricación permite circular una pequeña corriente. Por consiguiente, la *potencia estática* simplificada se puede describir mediante la siguiente expresión, donde  $I_{sub}$  es la corriente subumbral y  $V_{dd}$  es la tensión de alimentación:

$$P_{ESTÁTICA} = I_{SUB} \cdot V_{dd} \quad [\text{Ecuación 2.1}]$$

Por otro lado, la conmutación de los transistores causa la *potencia dinámica*, la cual es la mayor fuente de calor del procesador que se aproxima a la siguiente ecuación, donde  $C$  es la capacitancia,  $V_{dd}$  es la tensión de alimentación y  $f$  la frecuencia:

$$P_{DINÁMICA} = \frac{1}{2} C \cdot V_{dd}^2 \cdot f \quad [\text{Ecuación 2.2}]$$

Por consiguiente, la *potencia total* consumida se puede entender como el sumatorio de la potencia estática y la potencia dinámica [23].

$$P_{TOTAL} = P_{ESTÁTICA} + P_{DINÁMICA} \quad [\text{Ecuación 2.3}]$$

Se podría simplificar mediante la siguiente expresión que representa la cantidad de potencia transformada en calor:

$$P = P_{DINÁMICA} + P_{ESTÁTICA} = \frac{1}{2} C \cdot V_{dd}^2 \cdot f + I_{SUB} \cdot V_{dd} \quad [\text{Ecuación 2.4}]$$

Se puede apreciar, por lo tanto, que el término de la tensión de alimentación toma mayor influencia en la ecuación. Pero como Mudge [21] explica esa tensión es dependiente de la frecuencia, el cual describe además que la frecuencia máxima,  $f_{MAX}$ , que se puede alcanzar es la siguiente. En donde  $V_{th}$  es la tensión umbral de los transistores:

$$f_{MAX} \propto \frac{(V_{dd} - V_{th})^2}{V_{dd}} \quad [\text{Ecuación 2.5}]$$

Hoy en día a la frecuencia a la que se puede trabajar es elevada, y por lo tanto se puede conseguir mayor rapidez en el procesador, y a su vez mayor eficiencia, pero es la disipación del calor la que limita.

## 2.4 Fenómenos de transferencia de calor

Los fenómenos de transferencia de calor son importantes para poder entender lo que sucede al introducir un caudal de aire a una determinada temperatura y, además, comprender como se disipa el calor generado desde la propia plataforma al exterior.

Existen tres formas básicas de transferencia de calor: Conducción, convección y radiación [25].

Siendo la resistencia total a que se transfiere la suma de las tres resistencias:

$$R_t = R_{cond} + R_{conv} + R_{rad} \quad [\text{Ecuación 2.6}]$$

### 2.4.1 Conducción

La *conducción* se produce cuando entran en contacto dos o más regiones que se encuentran a diferentes temperaturas, las cuales tienden a igualarse. Este fenómeno se produce únicamente por contacto físico. La siguiente expresión describe la resistencia térmica por conducción [ $^{\circ}\text{C}/\text{W}$ ]:

$$R_{cond} = \frac{L}{k \cdot A} \quad [\text{Ecuación 2.7}]$$

Dónde  $A$  ( $\text{m}^2$ ) es el área de las regiones en contacto que indica además que, a mayor tamaño de lámina termo difusora del chip, menor será la resistencia.  $L$  (m) es la distancia entre las regiones que indica por otro lado, que cuanto mayor sea el grado de adherencia menor resistencia se obtiene. Por último,  $k$  ( $\text{W}/(\text{m} \cdot ^{\circ}\text{C})$ ) representa la conductividad térmica, que depende de las propiedades intrínsecas del medio que trasmite el calor de una región a otra y que indica los mejores materiales para la disipación de calor.

La conductividad térmica a 25°C del aire es de 0.023 W/(m·°C), por lo que adquiere un comportamiento térmico de aislante. En cambio, los materiales que se encuentran en la plataforma como el cobre, el cual tiene una conductividad térmica de 430 W/(m·°C) a 25°C, son unos excelentes conductores.

### 2.4.2 Convección

Una segunda forma de transferencia de calor es la *convección*, esta ocurre mediante el transporte de calor por medio del movimiento del fluido, ya que este fenómeno es más acusado en gases y líquidos. En nuestro caso va a tomar una especial relevancia ya que refrigeraremos con aire a diferentes temperaturas.

Por un lado, se da la *convección natural* que se debe a la diferencia de densidades o presiones que existen entre las regiones calientes y frías. Por otro lado, la *convección forzada*, que es provocada mediante la generación de turbulencias a través de un ventilador o bomba, o en nuestro caso con un secador con el que además del efecto de la convección forzada causado por el caudal de aire, se dispone de un aporte de calor.

La resistencia térmica por convección (°C/W) se puede expresar mediante la siguiente fórmula:

$$R_{conv} = \frac{1}{h \cdot A} \quad [\text{Ecuación 2.8}]$$

Donde  $A$  (m<sup>2</sup>) es el área de transferencia de calor y  $h$  (W/(m<sup>2</sup>·°C)) es el coeficiente de transferencia de calor convectivo, el cual depende de su grado de movilidad, es por ello por lo que es mayor en los gases que en los líquidos y a su vez que en los sólidos.

Se puede saber que el calor de un chip puede ser aliviado por el fenómeno que produce la convección natural en contacto con la atmósfera cuya resistencia se encuentra entre 15 y 240 °C/W. Pero si se aprovecha la convección forzada, por ejemplo, mediante un ventilador esta resistencia bajaría hasta aproximadamente 5 °C/W.

En cuanto al coeficiente de convección,  $h$ , se define como la cantidad de energía que se transporta en el fluido a distintas temperaturas. Dicho coeficiente es el indicador principal de la transferencia de calor, ya que indica el calor que se genera respecto del cambio de temperatura de un cuerpo. La [Ecuación 2.9] describe la transmisión de forma general [26].

$$Q = h \cdot A \cdot (T_S - T_F) \quad [\text{Ecuación 2.9}]$$

Donde,  $Q$  es el calor en potencia ( $W$ ),  $A$  es el área de disipación del dispositivo,  $T_s$  ( $^{\circ}C$ ) la temperatura del sólido y  $T_F$  ( $^{\circ}C$ ) la temperatura del fluido.

Por ello, es de interés analizar la influencia del coeficiente convectivo,  $h$ , cuando se produce un aumento en la velocidad del aire. Este coeficiente se suele determinar principalmente a través de correlaciones empíricas, obtenidas a partir de unos determinados experimentos, sin embargo, cada vez aumentan más los programas de ordenador con resoluciones numéricas según el caso.

Como se observa en la [Ecuación 2.9] si aumenta la convección, aumenta el coeficiente  $h$  y a su vez el flujo de calor. La convección forzada se evalúa mediante el parámetro adimensional denominado *número Nusselt* ( $Nu$ ), el cual lo representa la [Ecuación 2.10], donde  $L$  y  $k$  son la longitud característica y la conductividad térmica, respectivamente.

$$Nu = \frac{h \cdot L}{k} \quad \text{[Ecuación 2.10]}$$

Este número adimensional,  $Nu$ , depende de muchas variables entre ellas la geometría, la velocidad, temperatura de la pared, temperatura del fluido alejada de la pared y otros factores. Para simplificar los cálculos se utilizan parámetros adimensionales como el número de Reynolds ( $Re$ ), número de Prandtl ( $Pr$ ), número de Grashof ( $Gr$ ), etc. En casi todos los casos el número Nusselt se puede ajustar a la [Ecuación 2.11]:

$$Nu = cte \cdot Re^m \cdot Pr^n \quad \text{[Ecuación 2.11]}$$

Entonces, el *número Reynolds* influye directamente en el cálculo del *número Nusselt*, ya que mide la relación entre las fuerzas de inercia y las viscosas, y caracteriza el tipo de flujo (Laminar/Turbulento), por lo que el *número Nusselt* es directamente proporcional al *número de Reynolds*. Dicho número, por ejemplo, si se está en el caso de un espacio de sección cuadrada se puede calcular mediante la siguiente expresión:

$$Re = \frac{\rho \cdot V \cdot D_H}{\mu} \quad \text{[Ecuación 2.12]}$$

Dónde  $\rho$ ,  $V$ ,  $D_H$ , y  $\mu$  son la densidad del fluido, la velocidad característica del fluido, diámetro hidráulico a través de la cual circula el fluido y la viscosidad dinámica del fluido, respectivamente.

Por lo tanto, se puede concluir como se podía intuir que, al aumentar la velocidad del aire, aumentará el *número de Reynolds*, entonces aumentará el *número de Nusselt* y puesto que es directamente

proporcional con  $h$  ([Ecuación 2.10]), se confirma la idea de que al incrementar la velocidad del aire se incrementa la transferencia de calor.

### 2.4.3 Radiación

Por último, la radiación es la energía que irradian y absorben todo tipo de cuerpos de manera continua en forma de ondas electromagnéticas. Si el cuerpo está más caliente que su entorno emite más energía que la que recibe y viceversa en el caso contrario.

La resistencia térmica por radiación ( $^{\circ}\text{C}/\text{W}$ ) se puede expresar mediante la siguiente expresión:

$$R_{rad} = \frac{1}{\alpha_{rad} \cdot A} \quad [\text{Ecuación 2.13}]$$

Dónde  $\alpha_{rad}$  ( $\text{W}/(\text{m}^2 \cdot ^{\circ}\text{C})$ ) y  $A$  ( $\text{m}^2$ ) representan el coeficiente de radiación térmica y el área de transferencia de calor, respectivamente.

Este efecto tiene una velocidad mucho menor que en los fenómenos de conducción y convección y se puede despreciar en la mayoría de los casos.

## 2.5 Disipación de calor

En cuanto a la disipación de calor, las variables de mayor relevancia son dos. Por un lado, la cantidad de superficie que se tiene disponible para intercambiar calor con el medio, la cual viene diseñada por el fabricante. Por otro lado, la temperatura a la que se encuentra el medio con la que está en contacto con la placa y sobre la cual en sistemas ciberfísicos no tenemos influencia.

Entonces, cuanto mayor sea la diferencia de temperaturas entre el exterior y el procesador, mayor capacidad de disipación se obtendrá. Para aumentar esta se suelen introducir refrigerantes tales como aire, agua o aceites especiales. En la plataforma que se dispone y prácticamente en la totalidad de los procesadores, el procesador viene encapsulado con un material muy conductivo y una superficie mayor al silicio del procesador [6].

La disipación de calor se puede regir mediante un balance sencillo:

$$Entra = Sale + Acumula$$

$$Q_E = \underbrace{\frac{T_{\Delta P} - T_{\text{EXTERNA}}}{R_t}}_E + \underbrace{m\bar{c}_p \frac{dT_{\Delta P}}{dt}}_A \quad [\text{Ecuación 2.14}]$$

Donde:

- ❖  $Q_E$  (W) es el calor de entrada que se genera debido a la potencia eléctrica de la placa.
- ❖  $m$  (Kg) representa la masa total de la placa que intercambia calor, siendo el empaquetado del procesador el que disipa la mayor parte de calor.
- ❖  $\bar{c}_p$  ( $\frac{J}{^\circ\text{C}\cdot\text{Kg}}$ ) es el calor específico medio del material del empaquetado y la plataforma.
- ❖  $T_{\Delta P}$  ( $^\circ\text{C}$ ) es la temperatura en el procesador.
- ❖  $\frac{dT_{\Delta P}}{dt}$  representa la variación de la temperatura en el procesador.
- ❖  $T_{\text{EXTERNA}}$  ( $^\circ\text{C}$ ) es la temperatura del fluido que está en contacto con la placa.
- ❖  $R_t$  ( $\frac{^\circ\text{C}\cdot\text{s}}{\text{J}}$ ) es la resistencia térmica total a la disipación de la placa y determina la velocidad con la que se disipará el calor.

En la [Ecuación 2.14] por un lado, el término de salida se modela como el calor que se disipa desde el encapsulado al aire. Por otro lado, el término de acumulación representa el calor que se produce en el procesador y que se acumula en él. Un sistema o proceso ha alcanzado el estado estacionario si las variables que definen su comportamiento permanecen invariables respecto del tiempo, es decir, el término de acumulación es nulo.

Además, en el caso de usar refrigeración, como la de un ventilador, la resistencia térmica disminuirá por el aumento del coeficiente de transferencia de calor convectivo (como se comentaba en el apartado 2.4.2) y se conseguirá disipar mayor cantidad de calor, debido a la convección forzada.

Dado que se desconocen muchos de los datos relevantes, como el material del encapsulado, el calor específico medio o la masa, ya que no solo el encapsulado intercambia calor, no podemos usar esta ecuación. Es por ello por lo que realizaremos el ajuste de unos parámetros que caractericen los resultados obtenidos. Pero dicha ecuación, junto con los fenómenos de transferencia de calor ayudan a explicar el comportamiento y entender cómo se disipa la energía desde el interior de la placa al exterior.

## 2.6 Gestión en Linux de la temperatura y frecuencia

Para entender porque estamos realizando esta caracterización de los parámetros debemos entender antes como funciona *Linux* y como gestiona la temperatura y la frecuencia. Ya que *Linux* es el sistema operativo que se va a utilizar, él cual es ampliamente usado en los dispositivos *ciberfísicos* [6].

Linux realiza el control de la temperatura y es otro sistema el que se encarga de elegir la frecuencia de trabajo del procesador.

### 2.6.1 Gestión de temperatura

Para comprender como funciona, tenemos que entender cómo funciona el sistema térmico en *Linux*. Por un lado, en el hardware se encuentran los dispositivos mecánicos de enfriamiento y sensores. Por otro lado, se tiene el sistema del espacio de usuario con el que a través de una interfaz se puede modificar el sistema térmico, el cual se encuentra en el espacio de *kernel*. En este espacio de *kernel*, también se encuentran los dispositivos de enfriamiento y el *governor*, que es quién monitoriza la temperatura. Esta organización la podemos observar en el resumen de la Ilustración 2.2.

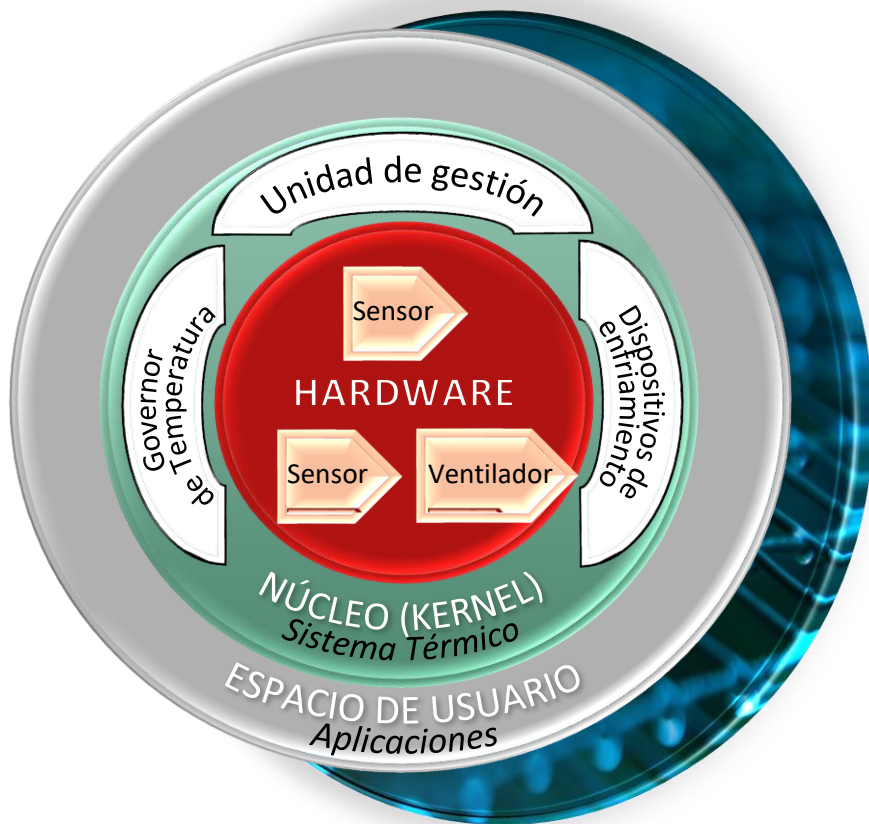


Ilustración 2.2 Esquema de los componentes del subsistema térmico en Linux.

El sistema que controla la temperatura se basa en la medición de la temperatura del dispositivo y en base a unos puntos de disparo activa distintos mecanismos de enfriamiento activo, el sistema que monitoriza la temperatura es conocido como *thermal governor*. Este sistema por consiguiente controla la velocidad de los ventiladores y la frecuencia máxima. Por lo tanto, se posibilita asignar dispositivos a un *governor* en la configuración de *kernel* personalizándolos para cada procesador. En la Ilustración 2.3 se puede observar un esquema sencillo de una de las zonas térmicas del *thermal governor*.

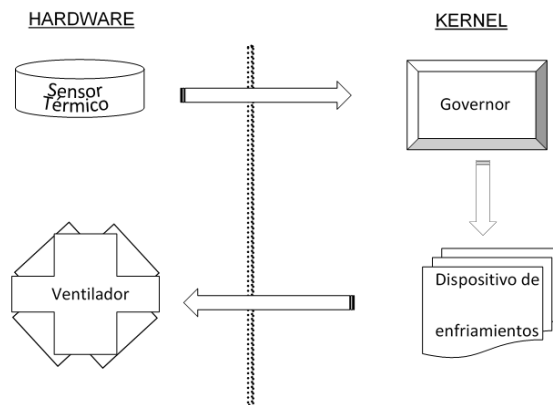


Ilustración 2.3 Esquema de una zona térmica del 'thermal governor'

### 2.6.2 Gestión de frecuencia

En cuanto a la frecuencia, es un *governor* diferente el que elige la frecuencia a la que trabaja el procesador. Esta frecuencia siempre se elige entre las que se encuentran disponibles, ya que estas pueden ser modificadas por el *governor* de temperatura en el momento en el que esta se excede, provocando ahogamiento térmico.

Los *governors* de frecuencia principales son:

- **Ondemand** es el que viene por defecto en la mayoría de los sistemas. Su forma de trabajar consiste en escoger la frecuencia según la carga de trabajo del procesador, es decir, las tareas por realizar, esta carga se calcula periódicamente. Por lo que, las frecuencias están limitadas por el *governor* de temperatura.
- **Performance** este governor elige siempre la frecuencia más alta entre las disponibles para mejorar el rendimiento.
- **Userspace** este sistema permite elegir al usuario la frecuencia del procesador, no es un *governor* como tal. Este sistema será usado para poder adoptar las simplificaciones de la [Ecuación 2.19].
- **Powersave** escoge siempre la mínima frecuencia disponible.



- **Intelligent Power Allocator** este es un *governor* que se ha recogido bibliográficamente diseñado por la empresa ARM [5], que utiliza un control PID para el uso en dispositivos móviles. Este PID sirve para realizar una estimación de la potencia disponible, en vez de un control directo sobre las acciones que modifican el comportamiento del sistema. Este es el *governor* más parecido al que se implantó en esta placa [6].

## 2.7 Identificación del sistema térmico

Para realizar la caracterización térmica la primera tarea consiste en conseguir un modelo matemático al que se puedan ajustar los datos obtenidos. Como se ha visto en el apartado de ‘Disipación de calor’, es posible obtener modelos de forma teórica [23], pero es realmente complicado obtener con precisión los valores de todas las variables que influyen.

Entonces, si partimos de la [Ecuación 2.4], y siendo que para modificar  $V_{dd}$  hay que modificar de la misma forma  $f$ , es decir, son variables dependientes, se puede simplificar la ecuación dejándola en función de  $f$  y un parámetro  $\omega$  que agrupa el resto de las variables.

$$P = \omega f^3 \quad [\text{Ecuación 2.15}]$$

A continuación, a partir del teorema de Taylor se linealiza la ecuación usando incrementos de la frecuencia con respecto a un punto de trabajo  $f_0$ . Por otro lado, se añade un término de retraso ( $\tau_d$ ), con el que se tendrá en cuenta el tiempo desde el momento que cambia la entrada del sistema hasta que afecta en la salida. Se obtiene la siguiente expresión, dónde ahora  $\delta$  es el parámetro que recoge el resto de las variables:

$$P = \delta f(t - \tau_d) \quad [\text{Ecuación 2.16}]$$

Seguidamente, sustituyendo en la [Ecuación 2.14], ya que el calor generado en el procesador ( $Q_E$ ) es el que se disipa, se obtiene la [Ecuación 2.17].

$$\delta f(t - \tau_d) = \frac{T_{\Delta P} - T_{\text{EXTERNA}}}{R_t} + m\bar{C}p\dot{T}_{\Delta P} \quad [\text{Ecuación 2.17}]$$

Pero como lo que interesa es la relación entre la temperatura del procesador y la temperatura externa, se fija la frecuencia ( $f$ ) y entonces, por el principio de superposición se puede omitir. A continuación, si agrupamos los términos simplificando  $\frac{1}{R_t}$  y  $m\bar{C}p$ , ya que es muy difícil obtenerlas de forma empírica, como  $K$  y  $\tau$ , respectivamente, se obtiene la siguiente ecuación:

$$K \cdot T_{EXTERNA} = K \cdot T_{\Delta P}(t) + \tau \cdot \dot{T}_{\Delta P}(t) \quad [\text{Ecuación 2.19}]$$

Por último, en el estudio del comportamiento de los sistemas se transforma esta expresión mediante Laplace con el objetivo de modelarlo pasando de la variable 't' a la variable compleja 's'. Por tanto, la ecuación de transferencia obtenida es la siguiente:

$$G(s) = \frac{T_{\Delta P}(s)}{T_{EXTERNA}(s)} = \frac{1}{1 + \tau \cdot s} \quad [\text{Ecuación 2.18}]$$

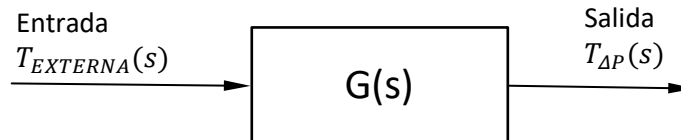


Ilustración 2.4 Función de Transferencia.

En este trabajo se pretende realizar un ajuste a una función de primer orden con retraso como se realizaba en el trabajo [6], en el que se pretendía ajustar la temperatura interna del procesador mediante la frecuencia de trabajo (ver Anexo B.1). Se puede observar que la [Ecuación 2.18] no dispone del término de ganancia ( $K$ ), pero debido a las características del sistema y a las condiciones de operación es previsible que sea necesaria. Además, se añade un término de retraso ( $\tau_d$ ), el cual posteriormente se estudiará si es necesario. Finalmente, la ecuación resultante es la siguiente:

$$G(s) = \frac{T_{\Delta P}(s)}{T_{EXTERNA}(s)} = \frac{K}{1 + \tau \cdot s} e^{-\tau_d \cdot s} \quad [\text{Ecuación 2.20}]$$

Por lo tanto, se quiere estudiar cómo se ajusta dicha función al sistema, y será a partir del análisis de los resultados, donde se pretende obtener los parámetros  $\tau_d$  (retraso ante un cambio en la entrada),  $K$  (ganancia del sistema) y  $\tau$  (tiempo de respuesta), relacionando la temperatura del procesador  $T_{\Delta P}(s)$  con la temperatura externa  $T_{EXTERNA}(s)$ . Dicha ecuación en sistemas de intercambio de calor es típicamente usada [32].

## 2.8 Identificación de los parámetros del sistema

El objetivo principal en este apartado es de ser capaz de, a partir de los resultados experimentales, identificar los parámetros de un sistema. Un requisito fundamental en un controlador es la obtención del modelo, obteniendo para ello los parámetros que describan la función de transferencia del controlador.

Cuando nos encontramos en un ámbito educativo las funciones de transferencia del controlador son siempre conocidas, pero la realidad es que en un sistema real está función no es conocida. Por ello,

mediante el proceso de identificación se quiere conseguir la obtención de la función de transferencia que se comporte de la forma más próxima a la real.

Esta identificación se realiza usualmente produciendo una excitación del sistema, conociendo la respuesta a dicha excitación, y ajustando posteriormente un modelo matemático. Esta excitación se suele realizar mediante: Una señal de escalón (cambio brusco) o una señal senoidal de amplitud constante y frecuencia variable.

La idea principal es realizar una identificación por medio de una función escalón, la cual es conocida como la identificación en tiempo.

### 2.8.1 Identificación de una respuesta a escalón

Esta es la señal más utilizada debido a la sencillez con la que se puede realizar, pero la realidad es que esto solo puede realizarse de forma aproximada, puesto que es imposible poder lograr un cambio a una variable de forma brusca en un tiempo infinitesimal.

El proceso de obtención de los parámetros del modelo dependerá de la naturaleza de estos, los cuales se pueden englobar en sistemas de primer orden y sistemas de segundo orden, ya que gran parte de los sistemas de orden superior pueden aproximarse por sistemas de segundo orden [28].

Este trabajo se va a centrar en el ajuste de un sistema de primer orden, ya que esa es la hipótesis que se ha realizado en el apartado 2.7.

Los sistemas de primer orden presentan una respuesta sobreamortiguada y no tienen sobreoscilación, además, son sistemas relativamente lentos. La función de transferencia de un sistema de primer orden, la cual contiene un solo polo, es la siguiente:

$$G(s) = \frac{K}{\tau s + 1} \quad [\text{Ecuación 2.21}]$$

En donde K es la ganancia del sistema y  $\tau$  es la constante de tiempo. A su vez la ganancia se calcula como:

$$K = \frac{c_1 - c}{r_1 - r} \quad [\text{Ecuación 2.22}]$$

En dónde,  $c_1 - c$  es la diferencia entre el valor inicial y final de la respuesta y  $r_1 - r$  es la amplitud del escalón producido. En la Ilustración 2.7 se puede observar cuales son dichos valores.

Por otro lado, la constante de tiempo ( $\tau$ ) también se calcula gráficamente, para ello lo que se hace es atender a la respuesta, siendo el valor de  $\tau$  el instante donde se ha alcanzado el 63,2% de su valor final. Por consiguiente, en la Ilustración 2.7,  $k$  es calculada de esta forma  $k = c + 0.632 * \Delta c$ .

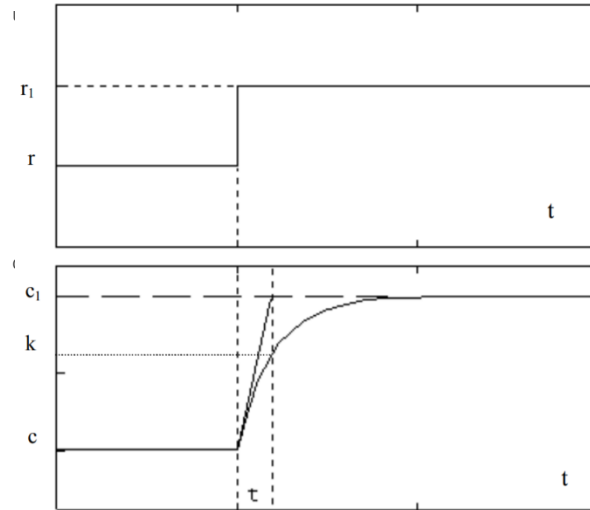


Ilustración 2.7 Primer orden con entrada escalón. [28]

Por otro lado, suele ser habitual que los sistemas presenten un tipo de retardo o desfase respecto a la señal de entrada. Para estos tipos de sistemas la función de transferencia es la siguiente, donde  $\tau_d$  es el retraso de tiempo que existe:

$$G(s) = \frac{K}{\tau \cdot s + 1} e^{-\tau_d \cdot s} \quad [\text{Ecuación 2.23}]$$

En la Ilustración 2.8 se puede observar el aspecto de este tipo de sistemas, y como se obtiene  $\tau_d$ . El resto de los valores se obtienen de la misma forma que en el caso de primer orden.

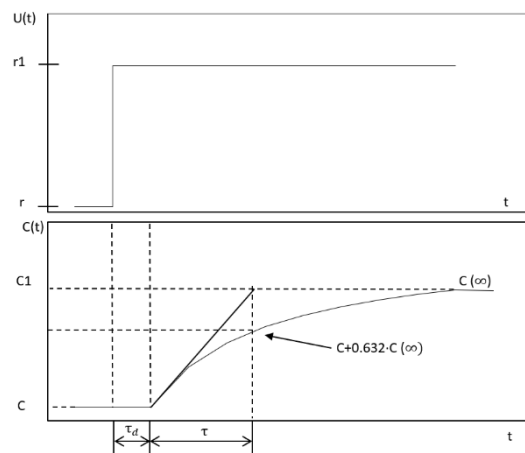


Ilustración 2.8 Primer orden con retraso con entrada escalón.  $C(t)$  vs tiempo.

# 3 PLATAFORMA DE CONTROL TÉRMICO DE BAJO COSTE

---

## 3.1 Requerimientos

Para poder controlar la temperatura ambiental del sistema es necesario "encerrar" la plataforma multiprocesador dentro de una estructura pequeña y aislada del exterior. Además de estos requisitos, será necesario poder variar de una manera controlada la temperatura dentro de la estructura y, por otro lado, ser capaces de introducir un caudal de aire. Las cámaras climáticas comerciales [36] cumplen los 4 requisitos anteriores pero su coste es demasiado elevado (13.500,00€), ya que otro de los objetivos de este TFG es reducir el coste lo máximo posible.

Entonces, por un lado, se requiere un sistema que sea capaz de elevar la temperatura del entorno de la placa en el interior de una estructura y, además, otro con el que podamos aportar un caudal de aire. Asimismo, se ha de escoger la forma de hacerlo y la instrumentación a emplear. Esta instrumentación debe ser económica y efectiva al mismo tiempo, también podría ser de interés que pueda generar cambios bruscos, lo cual provocaría escalones en la temperatura del entorno que facilitaría la creación de posibles modelos.

Por otro lado, es necesario un sistema de bajo coste donde podamos tener controlada la temperatura del entorno de forma homogénea y al mismo tiempo estresar térmicamente la placa, con el objetivo de poder obtener un modelo que tenga en cuenta el entorno. Además, otro de los objetivos es poder modelar un sistema refrigerado por aire, para obtener un modelo que tenga en cuenta tanto la temperatura del entorno, como el aporte o no de un caudal de aire controlado.

Por último, será necesario poder monitorizar la temperatura interna de la plataforma multiprocesador como del exterior en varios puntos para su posterior adquisición y análisis. Respecto al sistema externo de medida, se necesita un sistema que sea capaz de medir varios puntos de nuestra plataforma con el fin de poder obtener más información para el modelo térmico. Además, es interesante que tenga rapidez en el muestreo, exactitud y comodidad en la transmisión de datos. Asimismo, se necesita una dimensión adecuada del sensor del termopar y al mismo tiempo que sea económico.

En la Tabla 3.1 se puede encontrar un resumen tanto de los requisitos mínimos como de los recomendados de las partes de la plataforma de control térmico que se desea diseñar.

Tabla 3.1 Resumen de los requerimientos en el diseño de la plataforma de control térmico de bajo coste.

	Requisitos mínimos	Requisitos recomendados
<b>Estructura cerrada</b>	Bajo coste/ Espacio reducido aislado/ Capacidad de introducir caudal de aire/ Homogeneidad en la temperatura interior/ Fácil apertura para introducir la plataforma multiprocesador	Baja pérdida de calor
<b>Instrumentación temperatura-refrigeración</b>	Bajo coste/Capacidad de elevar temperatura interior y aporte de caudal	Capacidad de provocar un escalón en la entrada, variar el caudal y temperatura
<b>Sistema de medición temperaturas</b>	Bajo coste/ Capacidad de múltiples medidas/ Exactitud/ Dimensión del sensor/ Monitorización temperatura, adquisición y almacenamiento de datos	Comodidad/ Rapidez

### 3.2 Diseño físico

En la Ilustración 3.1 e Ilustración 3.2 se puede observar respectivamente un esquema de la plataforma de control térmico y una fotografía de la plataforma real, en la que el equipo de la derecha es el computador que recoge todas las mediciones que se realizan en la estructura de la izquierda.

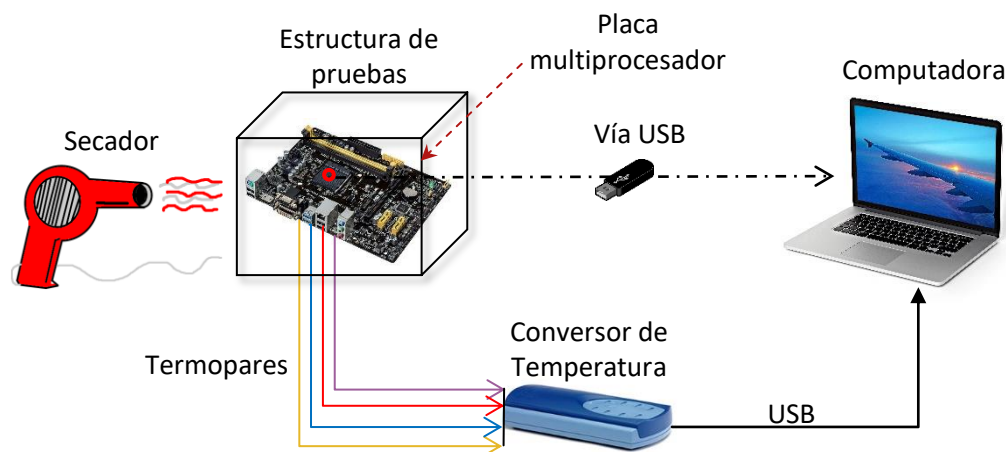


Ilustración 3.1 Esquema de la plataforma desarrollada. El círculo rojo corresponde al encapsulado por donde la placa disipa la mayor parte del calor.

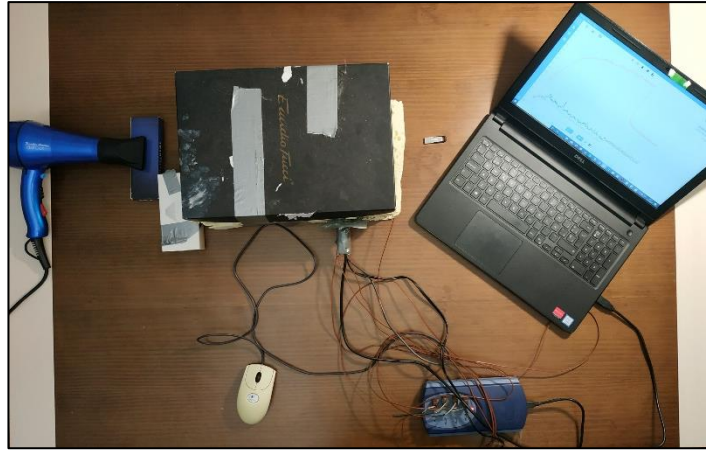


Ilustración 3.2 Fotografía de la plataforma desarrollada.

La plataforma que se desea diseñar consta de cuatro partes diferenciadas. En primer lugar, la instrumentación con la que se pondrá a prueba la placa mediante un estrés térmico y un sistema de aireación, en segundo lugar, el espacio donde se encuentra la placa la cual denominamos estructura de pruebas, la parte que mide la temperatura (interna) del sensor del procesador y finalmente, el sistema que mide la temperatura de forma externa, ajena al procesador.

### 3.2.1 Instrumentación refrigeración y temperatura

En la elección del sistema con el que se quiere estresar térmicamente la placa (elevando la temperatura del entorno), se consideró la opción de una resistencia que elevase la temperatura, por un lado, y por otro lado un ventilador como el que se implantan en ordenadores reales. Pero la necesidad de obtener una plataforma de control térmico de bajo coste y de conseguir un entorno homogéneo, hizo que se optase por el uso de un secador con el que se puede aplicar tanto aireación a temperatura ambiente como aireación a una temperatura elevada, por lo que se prevé conseguir un estrés térmico, una temperatura mucho más homogénea y un coste mucho menor.

Además, de esta manera se pueden conseguir cambios bruscos tanto en la temperatura como en la velocidad del aire en el interior de la estructura, haciendo que la entrada se produzca en forma de escalón de temperatura y/o caudal, lo que podría ser de ayuda a la hora de obtener modelos.

El secador tiene tanto un botón con 3 posiciones diferentes de calor como un botón para 3 posiciones posibles de caudal. En la Ilustración 3.3 podemos observar el secador utilizado en la plataforma. En cuanto a las posiciones de calor las denominaremos de la siguiente forma:

- “Calor bajo” Para la posición 1 (En la que la temperatura es aproximadamente la de ambiente).

- “Calor medio” Para la posición 2.
- “Calor alto” Para la posición 3.

En cuanto las posiciones del caudal las denominaremos tal que:

- “Caudal nulo” Posición 0 (En la que el secador esta apagado).
- “Caudal mínimo” Para la posición 1.
- “Caudal máximo” Para la posición 2.

Donde la magnitud del caudal en la posición media y alta se desconocen por falta de medios, pero no se considera para los modelos, ya que se pretende estimar el efecto mediante los resultados.



Ilustración 3.3 Secador usado en la plataforma.

### 3.2.2 Evolución de la estructura de pruebas

En vista a los requerimientos se optó en primera instancia por una caja de zapatos ya que cumple con la condición de que podamos introducir refrigeración y, además, su coste es muy bajo. La caja es por lo tanto la estructura de pruebas en la que se introduce la plataforma multiprocesador. A continuación, se decidió por donde se iban a introducir tanto los cables de la placa como los termopares. La localización y fijación de la placa es importante para que las pruebas sean reproducibles. Se realizaron pruebas, comprobándose que en una zona que no estuviera pegada a la pared ni muy cercana a la entrada obtendríamos un entorno más homogéneo para la placa. Estas localizaciones se pueden apreciar en la e Ilustración 3.4 e Ilustración 3.5.



Por otra parte, con el objetivo de que la influencia de la presión sea prácticamente despreciable, el aire que entre lo hará a una distancia de 10 cm por un agujero suficientemente grande y realice la función de entrada y salida del aire.

Se quiso observar el funcionamiento de la plataforma con la caja abierta y cerrada, es por ello por lo que se realizaron ensayos en la que se pone a prueba sin y con una fuerte carga de trabajo, en la que se observó el comportamiento. Por lo que se pudo concluir, que este sistema no retenía el calor todo lo bien que se quiere para las posteriores pruebas, además de que la transferencia de calor con el exterior era demasiado rápida y no se conseguía un sistema homogéneo. Algunas de estas pruebas las podemos encontrar en el Anexo A.1, en las que se puede apreciar que en apenas 30 segundos la temperatura descendía aproximadamente 13 °C.

Es por ello, que el sistema diseñado estaba lejos de servir como estructura de pruebas. Para solucionarlo, se ha realizado un encofrado con otra caja más pequeña mediante espuma de poliuretano[29], esta espuma es un aislante que se expande al entrar en contacto con el aire. El resultado puede verse en la Ilustración 3.4. Se ha comprobado entonces como aislar la caja con este aislante ha resultado bastante efectivo, suficiente para el objetivo que se quiere alcanzar. De la misma forma en el Anexo A.1 se pueden encontrar algunas de las pruebas realizadas, en las que se observa que en los primeros 15 minutos la temperatura había descendido 11 °C.

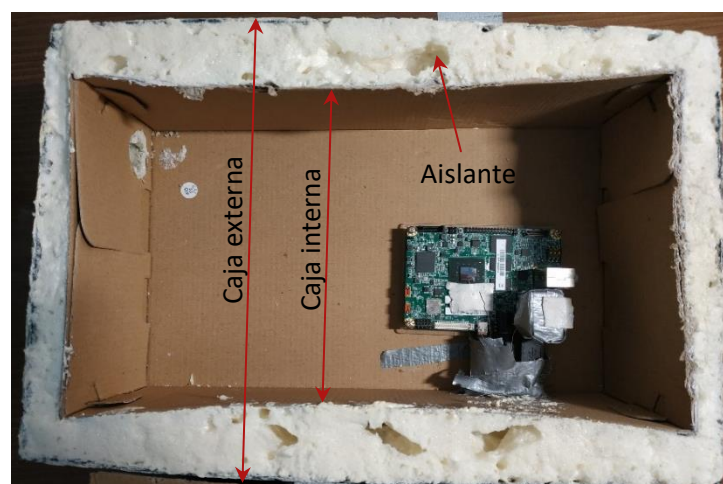


Ilustración 3.4 Estructura aislada con el encofrado de poliuretano.

En la Ilustración 3.5 se pueden observar en las fotografías el aislante de la tapa y la ubicación donde se realizaron los agujeros, tanto para la entrada del aire como la entrada de los cables de la placa y mediciones externas.



Ilustración 3.5 Estructura completa aislada.

En definitiva, se ha conseguido diseñar un sistema controlado de bajo coste en el que se puede poner a prueba la placa de forma real, pudiendo observar las limitaciones para obtener posteriormente una caracterización térmica más efectiva, ya que los dispositivos electrónicos no se encuentran en entornos en los que sea tan fácil disipar el calor, puesto que el procesador se encuentra en contacto con elementos que también disipan calor como, por ejemplo, las baterías.

### 3.2.3 Sistema de medición externo

Para la adquisición de los datos térmicos se necesitan dispositivos que capten la temperatura y otros que permitan su almacenamiento en un computador. En este caso, se tratan de termopares y un conversor analógico digital que transforma las diferencias en voltaje captadas por el termopar en valores binarios que pueden ser leídos vía USB.

La elección del tipo de termopar y del conversor es un punto importante para el posterior funcionamiento de la plataforma. En este caso se dispone de un sistema cuyo estudio del correcto funcionamiento está comprobado [7].

- ❖ El conversor elegido es el pico USB TC-08 [30], se puede observar en la Ilustración 3.6. Es capaz de tomar medidas cada 100ms y su precisión en la toma de medidas de temperatura es de  $\pm 2\%$ , asimismo, la interacción con la computadora se realiza de forma cómoda mediante USB, haciendo prescindible la alimentación externa. Además, cuenta con 8 canales por lo que permite tener hasta un máximo de 8 termopares midiendo temperaturas cada segundo.



Ilustración 3.6 Conversor USB TC-08

El funcionamiento del conversor consiste en el cálculo matemático de la medida real de temperatura, mediante la medida de voltaje que se obtiene del termopar y una medida de la temperatura ambiente en el interior del conversor. También realiza una estabilización de la señal.

- ❖ Las medidas se han llevado a cabo mediante termopares Tipo K miniatura de 0,5 m<sup>2</sup> de tamaño de punta. Se ha escogido este tipo de termopar ya que su mejor rango de trabajo se encuentra entre 20-90 °C, temperaturas sobre las que vamos a trabajar. Los termopares se unen a un conversor que transforma el voltaje. Este voltaje se produce por la diferencia de temperatura entre dos extremos denominados 'punto caliente' y 'punto frío', generados por un circuito con dos metales distintos. El conversor transmite la información a la computadora donde se almacena. En la Ilustración 3.7 se pueden observar las conexiones entre el conversor y los termopares.



Ilustración 3.7 Conexión conversor-termopares.

La ubicación de los termopares es importante para la obtención de datos precisos en localizaciones clave del procesador y la estructura. Junto con el profesor José María Marín Herrero, se han decidido cuáles eran los puntos clave de generación y disipación de calor. En la Ilustración 3.8 se muestra la ubicación de dónde están situados definitivamente los puntos de medición seleccionados.

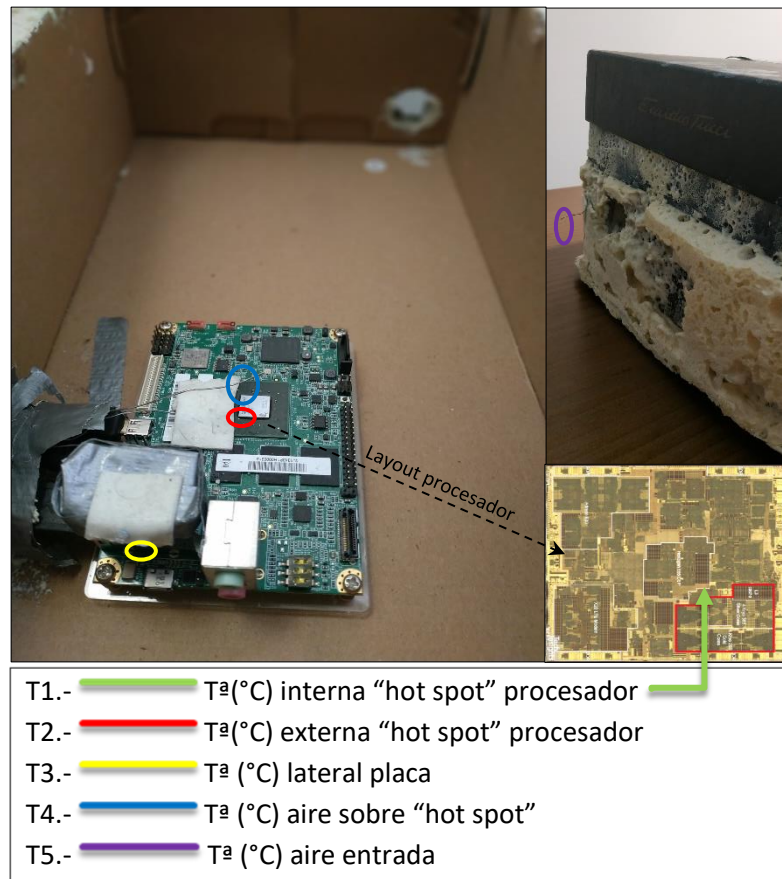


Ilustración 3.8 Puntos de medida escogidos.

Dichos puntos se han escogido para obtener una buena caracterización térmica de la plataforma multiprocesador. Los puntos elegidos se muestran en la Ilustración 3.8 y deben cumplir que:

- $T^a(^{\circ}\text{C})$  interna "hot spot" >  $T^a(^{\circ}\text{C})$  externa "hot spot" >  $T^a(^{\circ}\text{C})$  lateral placa/aire sobre "hot spot"
- $T^a(^{\circ}\text{C})$  lateral placa: Se mide esta temperatura para tener información de cuanto se calienta la placa al margen del 'hot spot' por el que se disipa la mayor parte del calor. En principio esta temperatura debería ser mayor que la  $T^a(^{\circ}\text{C})$  aire sobre "hot spot" ya que la capacidad de transmitir calor a través del aire es menor que por conducción en la misma placa.

### 3.3 Adquisición de datos térmicos

La plataforma desarrollada tiene otras dos partes bien diferenciadas. La parte de la plataforma que mide la temperatura interna del procesador y, por otro lado, la parte que mide la temperatura de forma externa e independiente al procesador.

En cuanto a la parte del sensor del procesador (medida interna), se realiza mediante el desarrollo de un programa que accede al sensor del procesador por medio de una biblioteca específica. En primer lugar, en el siguiente apartado (3.3.1) se describe este proceso, en segundo lugar, en el apartado 3.3.2 se explica cómo se han tratado los datos obtenidos mediante el conversor y, por último, se realiza la sincronización de estos.

#### 3.3.1 Medida interna

Para obtener la temperatura interna del procesador se debe acceder al sensor de temperatura del procesador gracias a la biblioteca *Im-sensors* [27]. Estas bibliotecas permiten recoger muchas características obtenidas de los sensores que tienen tanto la placa base como el procesador, dependiendo de su diseño. Gracias a estas librerías se han encontrado los datos del sensor del procesador.

Seguidamente, con el fin de poder capturar una secuencia de temperaturas al mismo tiempo que se ejecutan programas de mediciones, se usa un programa en el que con un hilo de ejecución lee temperaturas cada segundo y lo recoge en un fichero de texto. Además, se extraen al mismo tiempo la frecuencia a la que está trabajando como información adicional, lo cual podría ser útil posteriormente para comprobar limitaciones de la plataforma y aportar información relevante al respecto. Estos datos de igual manera se recogen en el mismo fichero de texto. Mediante este método las interferencias que se generan en el procesador son pequeñas.

Por otra parte, para poder realizar el análisis de los datos obtenidos es importante disponer de una plataforma donde poder tratarlos. El software de cálculo numérico *Matlab* ofrece las prestaciones necesarias entre las que se encuentran la manipulación de matrices, la representación de datos, la representación de funciones, etc. Pero, además, incluye herramientas como *Simulink* o paquetes de *Toolboxes* que pueden ser de utilidad a la hora de ajustar los modelos.

Por consiguiente, se creó un código en Matlab para tratar el fichero de texto el cual se muestra en el Anexo C.1.2. Se registran los valores de frecuencia y temperatura con una frecuencia de muestreo de 1 segundo.

### 3.3.2 Medidas externas

En cuanto la obtención de las medidas externas, la señal enviada por el conversor es recibida por el computador vía USB, pero es necesario un software para la adquisición de datos, el nombre del software utilizado es *PicoLog 6* [34]. Esto proporciona una interfaz visual y fácil de usar mientras se están ejecutando las pruebas. Lo que permite la recogida de los datos de las medidas externas, a la vez que se tiene información en tiempo real de que está sucediendo y de posibles fallos, además, permite tomar anotaciones que proporcionen información que pueda ser útil posteriormente.

Para el posterior análisis de los datos, de la misma forma que en la medida interna, interesa poder modular estos datos desde *Matlab*, para ello se han tenido que transformar los datos obtenidos en formato CSV mediante *PicoLog 6*. Esta transformación se ha realizado a través de un código el cual se encuentra en el Anexo C.1.3.

Cada segundo se puede tomar hasta un máximo de 8 medidas de temperatura, las mismas como termopares pueden estar conectados, estas medidas se recogen en un fichero CSV, que genera una fila con el fichero tal que:

```
00:00:00,"26,264","25,989","25,905","25,424","25,574","25,273","25,563","25,549"  
00:00:01,"26,268","25,993","25,905","25,420","25,589","25,262","25,567","25,541"  
00:00:02,"26,264","25,989","25,905","25,402","25,578","25,244","25,571","25,497"  
.  
.  
.  
01:01:16,"25,594","26,280","25,821","25,817","25,667","25,267","25,249","25,495"
```

Entonces, se leen los datos del archivo de texto en un array de celdas, separando los datos de los caracteres no numéricos. Luego, se crea la matriz de ceros en la que se introduce cada temperatura. A continuación, se crea la matriz de tiempo en el que se introducen en segundos los respectivos tiempos de cada medida de temperatura. Y finalmente, la posterior representación con la que se comprueba el funcionamiento.

### 3.3.3 Sincronización en Matlab

Una vez se ha realizado el código tanto para las medidas externas y la medida interna, se pasa a la representación conjunta de ambas.

En la Ilustración 3.9 se puede observar la representación tanto de las medidas externas como la interna. Pero como se aprecia, existe desincronización entre la medida interna y las externas. Esto es debido a que, en el proceso de ejecución de la plataforma, lo primero que se enciende son los termopares externos ya que interesa conocer el entorno antes y después de encender la placa. Es decir, el sistema de medición externo comienza a leer temperaturas almacenándose a través del software *PicoLog 6*, pero es en un momento dado cuando se enciende la placa y se puede comenzar a leer temperaturas con el sensor interno. Es por ello por lo que se observa ese error constante, debido a que la temperatura interna no comienza a medir en el segundo cero.

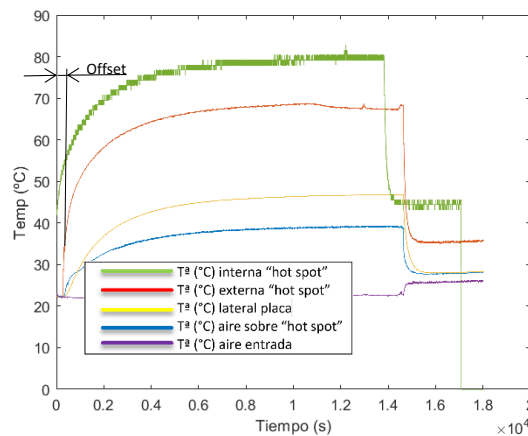


Ilustración 3.9 Problema de sincronización de datos.

Para solucionarlo, se inicia un cronómetro en el momento que se empieza a medir las temperaturas externas hasta el instante en el que se comienzan a leer temperaturas en el interior. Este intervalo de tiempo se convertirá en el punto de inicio del vector de la temperatura interna.

En la Ilustración 3.10 se observa el resultado, se puede comprobar como al inicio las curvas se ajustan bastante, pero a medida que el tiempo transcurre se puede apreciar discordancia entre ellas. Entonces, se deduce que a parte del problema anterior existe una discordancia generada por alguno de los sensores. Se ha comprobado tanto las mediciones externas como la interna. Al comprobar la temperatura interna, se ha observado que a pesar de que el tiempo de muestreo era de 1 segundo, la realidad es que varía décimas de segundo. Puesto que cada prueba puede durar hasta un máximo de 6 horas, ya que el error se va acumulando cada segundo, esto es más que suficiente para ser



significativo. Por ejemplo, suponiéndose un error constante de 0,04 segundos en una prueba de 1 hora el error es de 144 segundos.

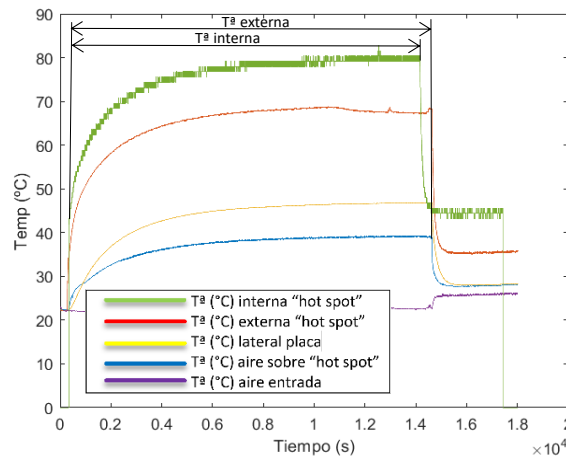


Ilustración 3.10 Sincronización sin tiempo de retraso.

Entonces, cuanto más dura la prueba menor error se obtiene a la hora de determinar el error constante, ya que para calcularlo se divide el tiempo real del cronómetro en el que el sensor interno está midiendo temperaturas, por el número de temperaturas que ha registrado, lo que da como resultado la frecuencia de muestreo real del sensor. Al intentar determinar la frecuencia real de muestreo, se aprecia que según la duración de la prueba la frecuencia de muestreo real varía, esto se puede deber a diversos factores como el estrés térmico que está teniendo en ese momento. Además, se puede comprobar como al principio la placa es más lenta y está por encima de la frecuencia real media, pero conforme pasa el tiempo es más rápida y la frecuencia de muestreo disminuye. Por consiguiente, cada prueba tiene un tiempo de muestreo real medio, lo que obliga a tener que tomar tanto el tiempo hasta que se empieza a leer temperaturas en el sensor interno, para subsanar el primer desfase, como el tiempo durante el que se leen temperaturas.

En la Ilustración 3.11 se puede observar como finalmente se ha conseguido la sincronización de los datos. Puesto que para la representación de los datos en *Matlab* se requieren vectores de la misma longitud, se tuvo que realizar una interpolación de los datos.



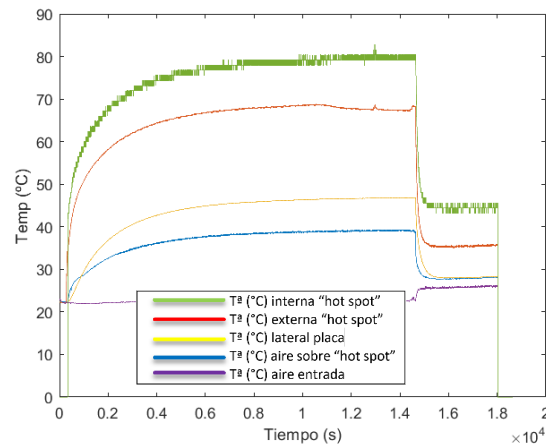


Ilustración 3.11 Datos sincronizados.

Por lo tanto, el fragmento del código de los datos externos no se tiene que modificar. En el código del Anexo C.1.4, se muestran los fragmentos del código del tratamiento de los datos internos con las modificaciones realizadas. En el que  $T_{sx}$  es la frecuencia de muestreo real media, esta se halla para cada prueba. Se debe calcular con las mediciones descritas anteriormente. Luego, se crea una matriz,  $T_{ss}$ , dicha longitud debe ser la misma que la obtenida con los datos de temperatura externa, ya que para poder representar los datos en *Matlab* es imprescindible.

Posteriormente, se realiza la corrección del tiempo de retraso hasta que se comienza a leer temperaturas, mediante el sensor interno. En este caso esta corrección se le ha denominado *offset*, el cual habrá que introducirlo en segundos. De la misma forma que antes con  $T_{ss}$ , la matriz de temperatura interna debe tener exactamente la misma longitud que la matriz de temperaturas externas.

Una vez se ha resuelto el problema del error constante, se realiza la interpolación de los datos de temperatura interna respecto a la matriz  $T_{ss}$ , que corresponde a la frecuencia real de muestreo, por último, se realiza la representación final. En la Ilustración 3.12 se muestra el funcionamiento general de la metodología usada para la resolución del problema de sincronización.

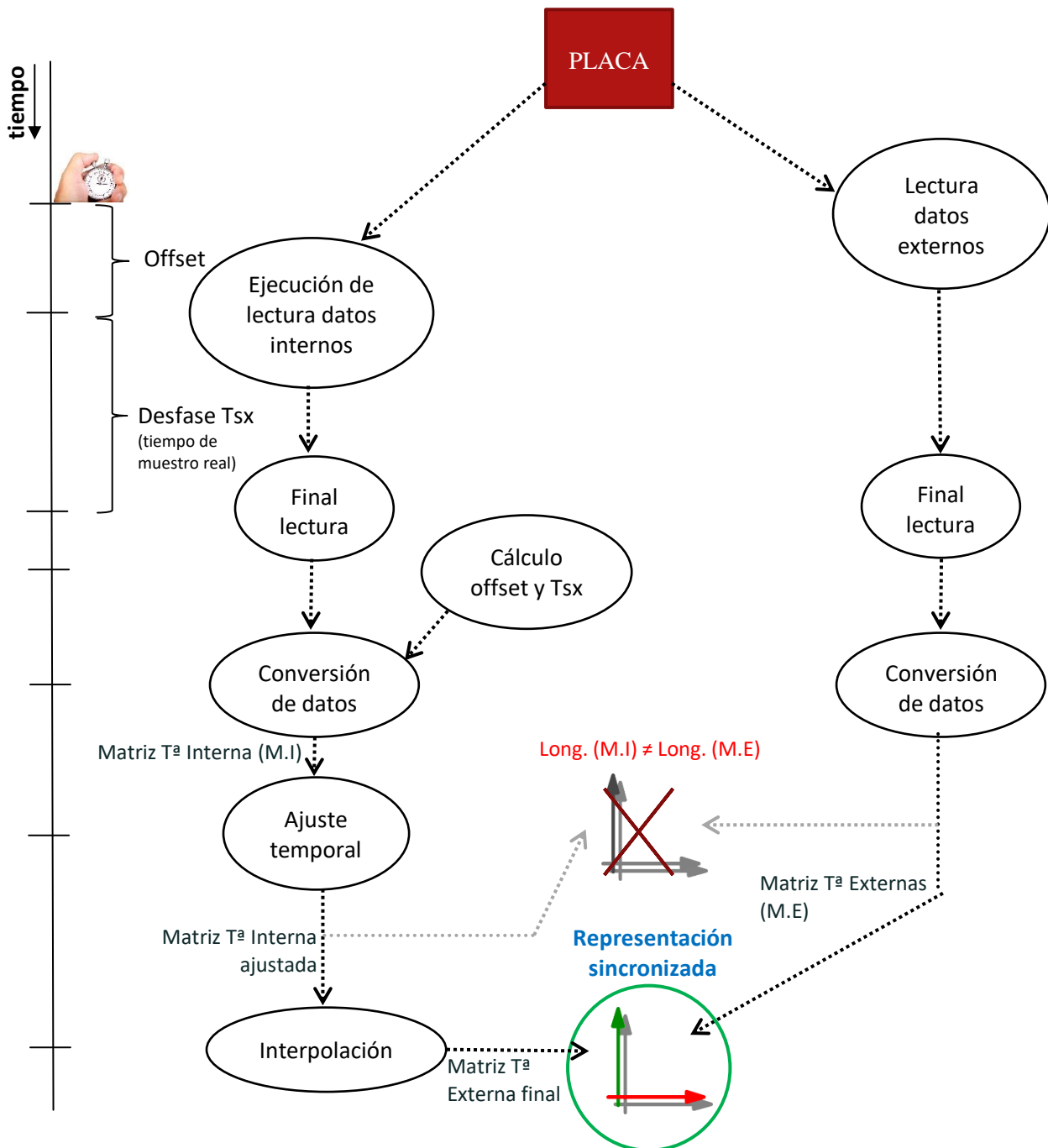


Ilustración 3.12 Esquema general de la metodología usada en la adquisición y sincronización de datos.

### 3.4 Ejemplo de uso

El proceso de uso de la plataforma de control térmico que se seguirá en cada prueba es el siguiente:

1. Preparación de la estructura y cerramiento de esta con cinta aislante, en el caso de realizar la prueba con la caja cerrada. Teniendo el agujero de entrada de aire cerrado también y el secador preparado a una distancia de 10 cm. Además, la placa debe estar preparada para ser

conectada a la corriente eléctrica y conectada a una pantalla con conexión HDMI desde la que podamos ejecutar las mediciones internas.

Por otro lado, el computador tiene que estar preparado con el conversor conectado vía USB, el software *PicoLog 6* abierto y los termopares en su posición correspondiente.

2. Se comienza a recoger temperaturas mediante el sistema externo de temperatura con *PicoLog 6*. En esta fase se está tomando la referencia de la temperatura ambiente antes de cualquier perturbación. Justo en el momento en el que se comienza a leer temperaturas se inicia el cronómetro con el que se obtendrá posteriormente la frecuencia de muestreo real y el *offset*.
3. Aproximadamente un minuto después, se enchufa la placa a la corriente eléctrica y se deja que se vaya encendiendo.
4. En cuanto la placa ha cargado la interfaz principal de Linux, se ejecuta el programa que lee la temperatura del procesador y a su vez, se lanza el fichero que fija la frecuencia. Al mismo tiempo, se anota cuanto tiempo ha transcurrido desde que se comenzó a leer temperaturas con la plataforma exterior, hasta este momento. Este tiempo corresponde al *offset* que hay que introducir en el código en *Matlab*.
5. Ahora, tanto el sistema interno como externo de medición de temperaturas está en funcionamiento, se espera a que el sistema se encuentre en estado estacionario, es decir, la transmisión de calor con el exterior sea igual a la que se genera.
6. Entonces, se pasa a realizar las correspondientes pruebas con y sin uso del secador. En el caso de realizarlo con secador, se retira rápidamente el tapón del agujero que corresponde a la entrada del aire y se enciende el secador en la posición deseada.
7. Cuando se ha alcanzado el punto al que queremos llegar se apaga tanto el secador si está enchufado, como la lectura de temperatura del procesador de la placa (a la vez que se anota el tiempo que ha estado leyendo temperaturas) y, por último, se para desde el computador la lectura de los datos de temperaturas exteriores con *PicoLog 6*.
8. Para finalizar, se guarda el fichero de texto en el que se han registrado tanto la temperatura del procesador como la frecuencia a la que estaba trabajando, introduciéndose en un pendrive que posteriormente se llevara al computador, en donde se analizarán mediante *Matlab*. De igual manera, desde *PicoLog 6* se guardan los datos en el formato de fichero CSV, para el posterior análisis en *Matlab*.

# 4 CARACTERIZACIÓN TÉRMICA

---

Este capítulo, por un lado, describe los experimentos realizados para verificar que la plataforma es viable y útil y, por otro lado, los experimentos necesarios para modelar la placa, además de servir como herramienta para ayudar a entender el comportamiento de la placa en función de las condiciones del entorno. Para ello, inicialmente se realizó un análisis de los posibles experimentos a ejecutar.

## 4.1 Diseño de los experimentos

Al realizar el diseño de los experimentos se balanceó cuidadosamente tanto el número de estos como las variables a analizar. Por un lado, un número muy alto aumentaría demasiado el tiempo de experimentación, y por otro, realizar pocos podría perder información útil. Por lo tanto, se decidió modificar el siguiente conjunto de variables:

Experimentos según el número de *cores*, con y sin carga de trabajo, ya que los procesadores acostumbran a trabajar con una fuerte carga, lo que provoca una subida considerable en la temperatura de la placa. Por consiguiente, resulta de interés conocer y comparar este comportamiento. Por otro lado, para poder ser capaces de comparar este sistema y la adaptación a otro tipo de ambientes resulta de interés realizar experimentos tanto con la estructura cerrada como abierta.

Entonces, se dispone de las siguientes combinaciones de experimentos, las cuales se muestran en la Ilustración 4.1, según el número de *cores*, la estructura abierta o cerrada, el caudal introducido y la temperatura del aire.

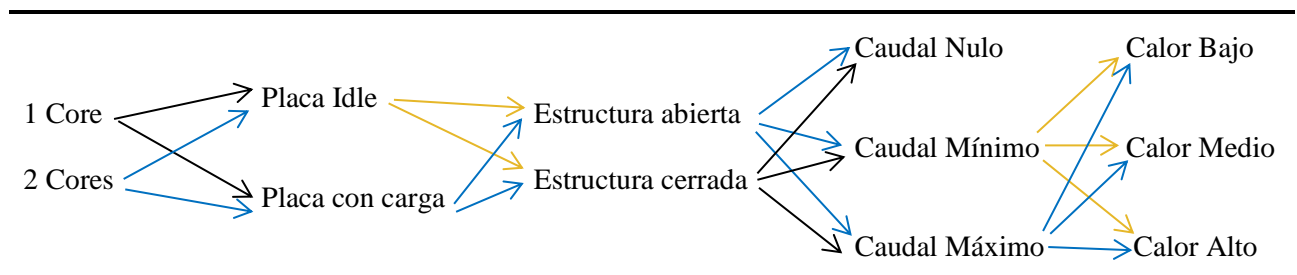


Ilustración 4.1 Esquema de las posibles combinaciones de los experimentos.

En los experimentos en los que se trabaja con caudal de aire a elevada temperatura, podría peligrar la integridad del secador llegando a causarle daños, además de la energía consumida durante el experimento. Por dichos motivos es importante elegir e ir observando los resultados obtenidos con el fin de descartar posibles experimentos innecesarios. Asimismo, se deja para trabajo futuro por limitaciones de tiempo el análisis en función del número de *cores*.

## 4.2 Resultados obtenidos

En dichas pruebas es muy importante fijar la frecuencia de trabajo para posteriormente poder tratarla como una constante, como se comentaba en el apartado 2.7, por ello en cada prueba se ha comprobado el correcto funcionamiento del ejecutable, tal como se muestra en la Ilustración 4.2.

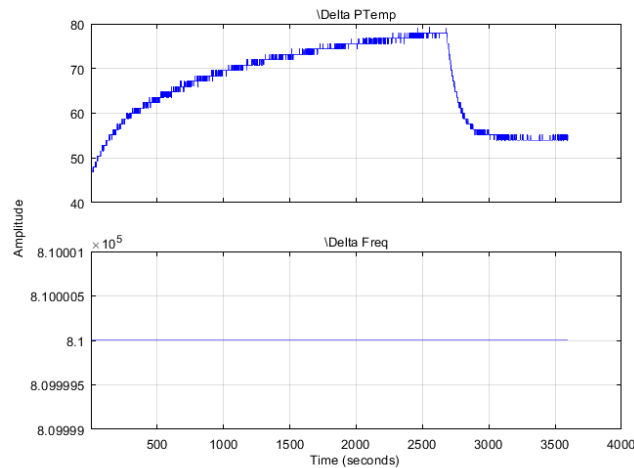


Ilustración 4.2 Frecuencia fijada.

A continuación, en la Tabla 4.1 se muestra un resumen en las que se recogen los datos de mayor relevancia en estado estacionario, según la posición de calor y la posición del caudal. En dicha tabla se muestra información de las gráficas del momento en el que la temperatura se ha estabilizado, aunque en algunos casos no lo llegamos a alcanzar por el elevado tiempo que es necesario tener en funcionamiento el secador. Estas gráficas se pueden encontrar en el Anexo C.2, en ellas se puede observar y comprender mejor la evolución.

Por un lado, se pueden apreciar los resultados con la estructura abierta. Por otro lado, los resultados con la estructura cerrada. Además, se realizaron pruebas con y sin, *Idle*, carga de trabajo. Estas pruebas se realizan para diferentes posiciones de calor y caudal. A medida que se realizan pruebas, se desprecian otras que se deduce que vayan a ser irrelevantes como, por ejemplo, las realizadas con *Caudal alto*.

Tabla 4.1 Tabla resumen en estado estacionario de algunos de los experimentos. Los símbolos tienen los siguientes significados: 'X' la placa no soporta el estrés y se apaga, 's' no puede haber medida por las condiciones de operación y '-' no se llega a alcanzar el estado estacionario en el tiempo de operación del experimento.

	Estructura cerrada									Estructura abierta		
	Placa Idle					Placa con carga				Placa Idle		Placa con carga
	Caudal Mínimo			Caudal Máximo		Caudal Mínimo			Caudal Mínimo	Caudal Máximo	Caudal Nulo	
Temperaturas	Calor Bajo	Calor Medio	Calor Alto	Calor Bajo	Calor Medio	Calor Bajo	Calor Medio	Calor Alto	Calor Medio	Calor Bajo		
'Hot spot' interna sin caudal (°C)	79,8	81	78,6	78	78,6	X	X	X	65,4	67,8	66,6	82,1
'Hot spot' interna con caudal (°C)	44,4	-	X	54,6	-	59,9	87,5	X	-	49,8	46,2	s
$\Delta T$ con/sin caudal 'hot spot' (°C)	35,4	-	6,6	23,4	-	-	-	-	-	18	20,4	s
Aire de entrada	25,7	43,7	60,5	32,8	53	29,2	43,8	57,6	48,8	32,5	31	s
Ambiente (°C)	22,5	23,2	23,6	27,6	28,2	26	23,1	22,8	29,5	30,8	28,6	22,2
Tiempo total experimento (min)	300	284	168	83	73	194	184	24	59	45	50	186

Para cada prueba se refleja la lectura de la temperatura del termopar en estado estacionario de 4 medidas. En primer lugar, se muestra la temperatura del 'hot spot' interna del procesador antes de que ocurra cualquier entrada de aire si es que la hay. En segundo lugar, se indica la temperatura del 'hot spot' después de que un determinado caudal de aire se haya introducido a su respectiva posición de calor (si es que se ha introducido aire). En tercer lugar, se muestra la temperatura del aire que entra a través del secador, si es que entra aire. Por último, se refleja la temperatura ambiente de la habitación en el tiempo en el que se realizó el experimento. Además, se muestra el tiempo de ejecución del experimento y se indica la diferencia entre las dos primeras medidas, la cual refleja la capacidad de disipación del secador a unas determinadas condiciones.

La Tabla 4.1 únicamente muestra información del estado estacionario, pero el estado transitorio es fundamental para entender cómo se comporta el sistema. El estado transitorio depende de las condiciones iniciales, caracterizando la evolución temporal desde dicho estado hasta el régimen permanente. Este estado de transición se aprecia en las gráficas de cada uno de los experimentos.

### 4.2.1 Influencia estructura abierta-cerrada

Las principales conclusiones extraídas de estos experimentos son las siguientes:

- ❖ Durante los experimentos se observa que **la temperatura final con la estructura abierta es muy inferior comparada con la de la estructura cerrada**. Esto es debido a que con la estructura abierta la diferencia de temperaturas entre la temperatura del fluido ( $T_f$ ) y la temperatura del sólido ( $T_s$ ) aumenta, aumentando por lo tanto la transmisión de calor como se observaba en la [Ecuación 2.9]. En la Ilustración 4.4 e Ilustración 4.3 se pueden comparar los resultados entre una prueba con estructura abierta-cerrada con *Caudal mínimo* y *Calor medio*.
- ❖ En dichas pruebas se aprecia cómo **no es totalmente cierto que se alcance el estado estacionario**. Esto ocurre tanto con la estructura abierta como cerrada para las posiciones de *Calor medio* y *alto*. Es debido a la alta temperatura con la que entra el aire, el calor se transmite a la placa mediante los fenómenos de transferencia de calor, pero en este caso la conductividad térmica del aire es pequeña debido a su coeficiente de conductividad térmica, por lo tanto, tarda más en transmitirse que por convección.

Además, dichas pruebas se ejecutan durante horas haciendo que el secador se sobre caliente, provocando un pequeño aumento en la temperatura de entrada. Por otra parte, la radiación adquiere influencia a elevadas temperaturas, por consiguiente, la estructura se va calentando haciendo que se desprenda calor trasmitiéndose y afectando a la temperatura del procesador.

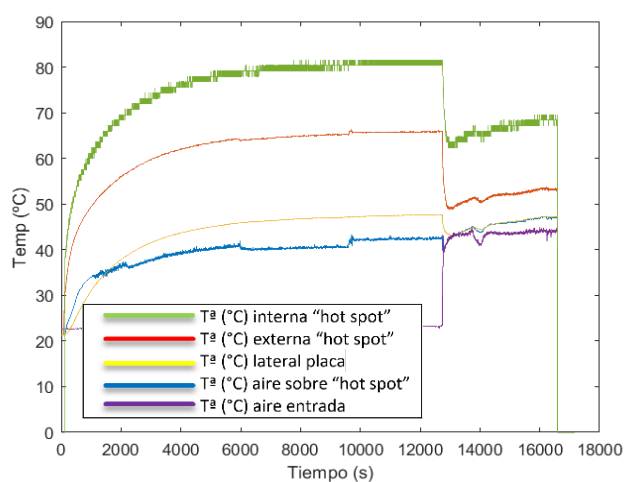


Ilustración 4.4 Estructura cerrada. Caudal mínimo y Calor medio.

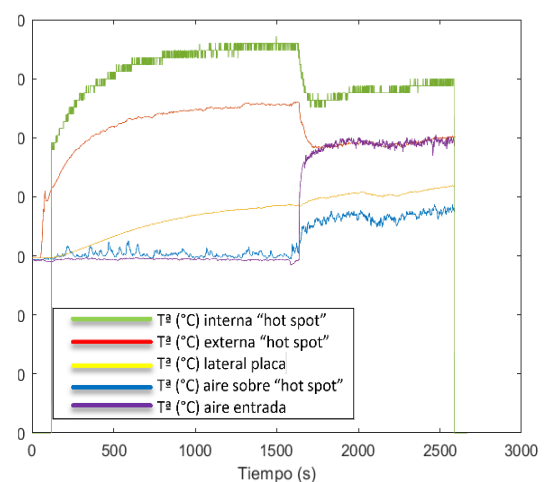


Ilustración 4.3 Estructura abierta. Caudal mínimo y Calor medio.

- ❖ La placa con carga de trabajo con la estructura cerrada y sin refrigeración se acaba apagando automáticamente. En las pruebas con una fuerte carga de trabajo se ha concluido que, con la estructura cerrada, la temperatura del entorno de la placa provoca que no soporte el estrés térmico y a los 5 minutos se acabe apagando, como se puede apreciar en la Ilustración 4.5. En cambio, en la Ilustración 4.6 se puede observar la realización de 3 repeticiones de las 7 iteraciones de multiplicación de matrices que se realizan en la placa, y cómo se comporta la misma en este entorno con la estructura abierta, concluyendo que en este entorno sí se llega a disipar suficiente a través del aire. Por lo tanto, para poder trabajar con una fuerte carga de trabajo con la estructura cerrada es necesario la refrigeración por aire.

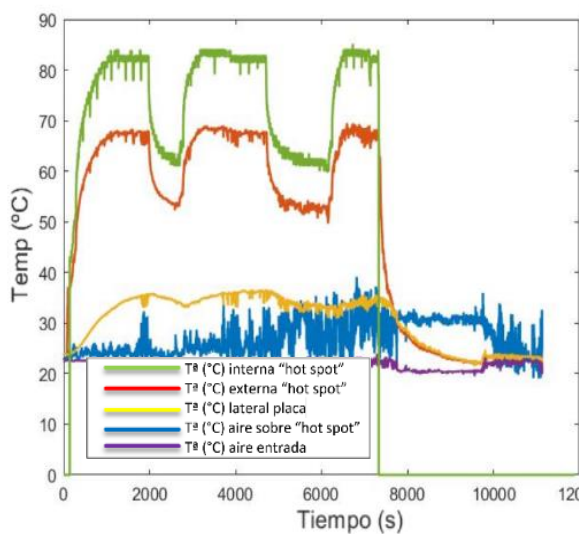


Ilustración 4.5 Estructura abierta. Placa con carga. Caudal nulo.

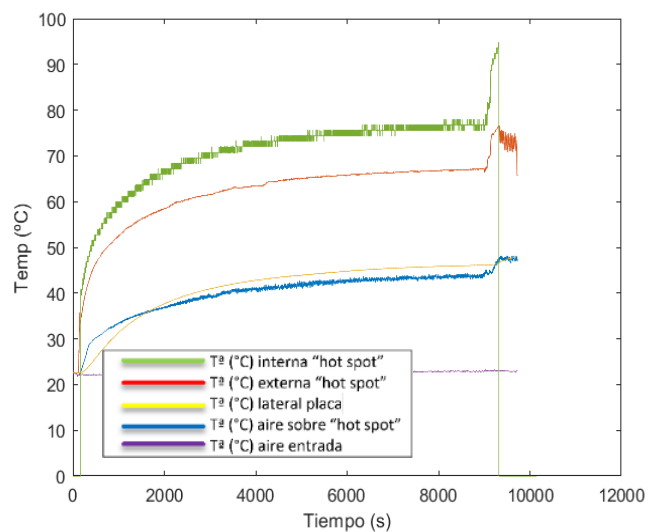


Ilustración 4.6 Estructura cerrada. Placa con carga. Caudal nulo.

#### 4.2.2 Influencia de la temperatura del entorno-Caudal aire

- ❖ A **mayor temperatura del aire** introducido **menor cantidad de calor se puede disipar** de la placa. Esto se puede observar tanto en la Tabla 4.1 como en las ilustraciones del Anexo C.2, algo lógico y esperado como se podía predecir con la [Ecuación 2.9].
- ❖ El efecto por **convección es de gran importancia** con estas magnitudes de caudal, a pesar de la elevada temperatura del aire. Si se toma como ejemplo la prueba de la Ilustración 4.4. Al introducir aire a 43,7°C en consecuencia se está consiguiendo retirar una gran cantidad de calor. Este resultado sorprendió bastante, ya que estamos elevando la temperatura del entorno de la placa en 20,5 °C, y a pesar de ello se consigue disminuir la temperatura en el 'hot spot' del procesador 18°C en 4 minutos.



- ❖ Los **resultados no son comparables** de forma inmediata. Ya que cuanto mayor es la temperatura ambiente, la temperatura del aire del secador aumenta en consecuencia. Esto es lógico ya que el secador aspira el aire del entorno de la habitación, por lo tanto, se ve claramente influido.
- ❖ **Con *Calor alto* y la estructura cerrada la placa no disipa el calor necesario y llega a apagarse** automáticamente para no dañarse. Se puede apreciar en la Ilustración 4.7, cuyo aire introducido se encuentra aproximadamente a 60,5 °C. Aunque se puede observar como en el primer minuto y medio se consigue refrigerar 6,6 °C, causado por el efecto de la convección de la misma forma que sucedía antes. Pero al encontrarse en un espacio cerrado, en poco tiempo se homogeniza el espacio interior haciendo que la temperatura se eleve tendiendo a un estado estacionario. En cambio, en este caso con *Calor alto* no se llega a alcanzar el estado estacionario y la placa se apaga.

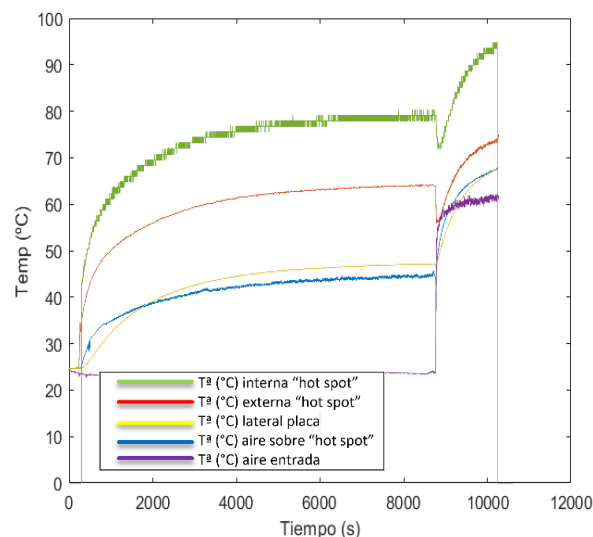


Ilustración 4.7 Estructura cerrada. Caudal mínimo y Calor alto.

- ✓ La temperatura del entorno de la placa tiene gran influencia en la disipación de calor. Pero, se ha podido observar que, en un **entorno con temperatura baja sin refrigeración**, la placa **disipa peor** que en un **entorno con la temperatura elevada pero refrigerada por aire**, esto es debido al efecto que tiene el aumento del coeficiente convectivo ( $h$ ), al aumentar la velocidad del aire.

Por ejemplo, esto se aprecia en la Ilustración 4.8 con la estructura abierta, la temperatura del entorno es la de ambiente y la temperatura en la que se estabiliza el procesador es de 66,6 °C. En cambio, en la Ilustración 4.4 con la estructura cerrada a 43,7 °C (modificado por el aire introducido en la posición *Calor medio*) la temperatura se estabiliza en 62,9 °C. Consiguiéndose por lo tanto reducir la temperatura 3,1 °C más que con la estructura abierta a baja temperatura.

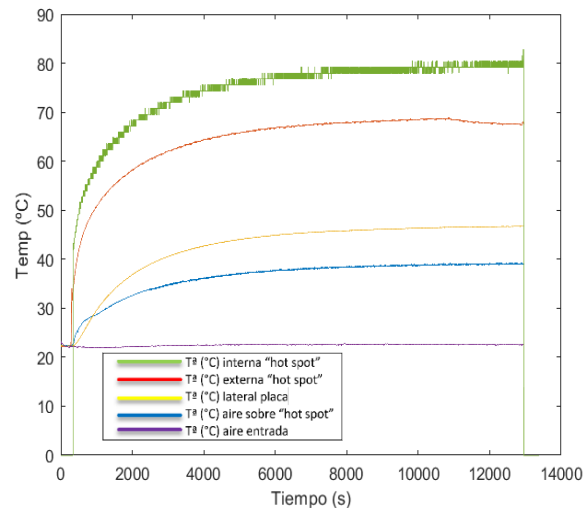


Ilustración 4.8 Estructura abierta. Caudal nulo.

En concreto, en los centros de datos en los que se requiere un enfriamiento continuo, se podría reutilizar el aire que enfría los discos duros en el enfriamiento de los procesadores, los cuales trabajan a una mayor temperatura, logrando de esta forma realizar un aprovechamiento energético. Por otro lado, se podría llegar a intentar buscar un óptimo entre la temperatura y el caudal de entrada para que se pueda llegar a refrigerar el sistema de la forma más efectiva y económica posible.

En conclusión, si se consigue caracterizar el sistema térmico teniendo en cuenta tanto la temperatura del entorno como la aportación de aire de refrigeración, se podría prever mejor el comportamiento del procesador y obtener un mejor control. Logrando de esta forma un aumento en la eficacia del dispositivo electrónico o la reducción del coste en refrigeración. Además, muchos sistemas no pueden contar con refrigeración mecánica por motivos de espacio y económicos, lo que podría ser de gran interés en este tipo de dispositivos como, por ejemplo, en los dispositivos móviles.

#### 4.2.3 Control con estrés de la placa

De forma paralela, se realizaron pruebas con el control PID instalado en la placa, en ella se realizaron simplificaciones notables en la función de transferencia que describe el sistema, en este control se regula la frecuencia de uso de la placa multiprocesador para controlar la temperatura interna, sin tener en cuenta el entorno. Pablo Hernández en su proyecto de fin de carrera [6], comprobó que se había mejorado la eficacia de la placa implementando un control PID, pero este sistema se encontraba en un entorno sin ningún estrés térmico, por ello, se quiere comprobar la seguridad de este a través de la plataforma diseñada.

La realización de los experimentos se ha llevado a cabo con la estructura cerrada, sin la introducción de un caudal de aire y una fuerte carga de trabajo, además, no se fijó la frecuencia de trabajo con el objetivo de que sea cada control quien la regule y poder apreciar así las diferencias.

En los resultados, se podría predecir que en ambos casos la placa se apagaría de la misma forma que ocurría con anterioridad en los experimentos realizados con una fuerte carga de trabajo y la estructura cerrada, y efectivamente así ocurrió. Estos resultados pueden observarse en el Anexo C.3, en los que con el control por defecto en tan solo 45,4 segundos la placa se apaga. Pero, aunque de igual manera la placa se acaba apagando, usando un control PID con los parámetros de control correspondientes y una temperatura objetivo de 85 °C, la placa permanece en funcionamiento 246,8 segundos.

Por lo tanto, se puede concluir que el control PID que se implementó en la placa no solo es más efectivo, sino que es más seguro que el instalado por defecto. Por lo que se ratifica que obteniendo un modelo térmico experimental se puede mejorar el control, además, teniendo en cuenta la temperatura del entorno y la refrigeración, se podría predecir las posibles situaciones actuando previamente y obteniendo un control más seguro y eficaz.

### 4.3 Análisis de resultados con Matlab y determinación de parámetros

En primer lugar, mediante la *Toolbox* de identificación de sistemas dinámicos que ofrece Matlab se han lanzado, a través de la operación '*Transfer Function Models*', diferentes estimaciones. Se ha obtenido un buen ajuste como se observa en la Ilustración 4.9, en donde el porcentaje de coincidencia es del 95,7%. La función de transferencia obtenida se muestra en la [Ecuación 4.1].

$$G(s) = \frac{-0.3795 s^4 - 0.03948 s^3 - 0.006278 s^2 - 0.0001328 s + 4.605e-09}{s^5 + 0.1315 s^4 + 0.01929 s^3 + 0.0008806 s^2 + 1.153e-05 s + 3.747e-12} \quad [\text{Ecuación 4.1}]$$

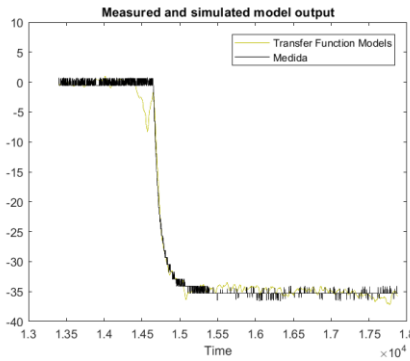


Ilustración 4.9 Ajuste para Calor bajo con Transfer Function Models

A pesar de haberse obtenido un buen ajuste la función es muy poco consistente, es decir, carece de sentido físico, ya que es un modelo tan complejo que no podría llegar a generalizarse.

Por lo tanto, se ha procedido a realizar un análisis tal como se exponía en el apartado 2.7 a partir de métodos gráficos, con el objetivo de obtener una función más próxima a la realidad.

Para ello, se ha procedido en primer lugar a comprobar que el sistema es totalmente sobreamortiguado, como se aprecia en la Ilustración 4.10, por lo que es razonable tratar de aproximarlos por un sistema de primer orden.

A través de las correspondientes salidas medidas, como consecuencia de la excitación aplicada en la entrada, por un lado, se calculan las constantes a través del método gráfico expuesto en el apartado 2.8.1. Estos modelos los denominaremos como 'Manual'. Se procede entonces al cálculo de las constantes de tiempo ( $\tau$ ), la ganancia del sistema ( $K$ ) y el retraso de tiempo ( $\tau_d$ ) que existe, con el objetivo de calcular la función de transferencia de la [Ecuación 2.20].

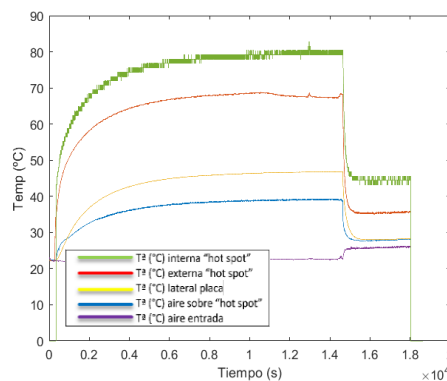


Ilustración 4.10 Estructura cerrada.  
Caudal mínimo y Calor bajo.

.

La señal de entrada en forma de escalón es proporcionada por el termopar que mide la temperatura del aire de la entrada, siendo la salida la medida recogida por el termopar interno del procesador. Esta última se identificará mediante la leyenda 'Medida'. En la Ilustración 4.12 se puede apreciar dichas señales, pero será la parte en la que se produce la excitación donde realizaremos el cálculo del modelo, ya que el método de cálculo así lo requiere, además, previamente en el código de Matlab se les ha restado el valor inicial antes de producirse el escalón para partir de condiciones nulas. En la Ilustración 4.11 se puede observar la señal final a modelar.

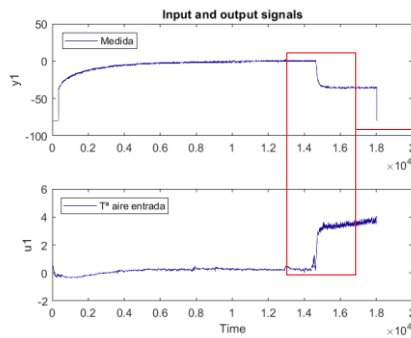


Ilustración 4.12 Señales de entrada y salida.

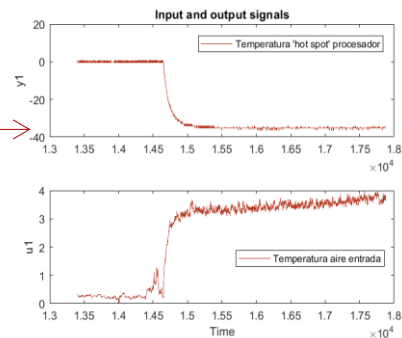


Ilustración 4.11 Señal de entrada y salida a modelar.

Ahora, una vez calculado manualmente los parámetros de las funciones de transferencia, se ha utilizado *Matlab* para mejorar dichas funciones. Observando los valores obtenidos se puede acotar en *Matlab* los rangos en los que se obtiene un resultado razonable. Así, de esta forma y a través de la operación 'Process Models' el modelo que se calcula es más preciso que el obtenido manualmente, a la par que se acerca a un modelo más real. En las gráficas se identifica mediante la leyenda 'Toolbox'.

En la Ilustración 4.13, Ilustración 4.14 e Ilustración 4.15 se pueden observar los ajustes realizados por ambos métodos a partir de las gráficas obtenidas con los experimentos con caja cerrada, la placa Idle y caudal mínimo. Por un lado, estos se ajustan a un modelo de primer orden ([Ecuación 2.20]) obteniendo las funciones denominadas como 'Manual' y 'Toolbox' y, por otro lado, a partir de la *Toolbox* de Matlab se obtienen expresiones mediante la adición de polos y ceros con el objetivo de investigar la función inicial propuesta. Con ello, se ha observado que mediante la adición de un cero el ajuste es más preciso, sobre todo cuando existe acumulación de calor. Esta expresión en las gráficas se identifica mediante la leyenda 'Toolbox más  $\tau_z$ ' y la define la siguiente función de transferencia ([Ecuación 4.2]), donde  $\tau_z$  es el conocido como la constante de tiempo cero:

$$G(s) = \frac{K \cdot (1 + \tau_z \cdot s)}{(\tau \cdot s + 1)} e^{-\tau_d \cdot s} \quad [\text{Ecuación 4.2}]$$

Por otro lado, se observa que el retraso,  $\tau_d$ , en la función ajustada a un primer orden o bien es irrelevante o no proporciona un buen ajuste al modelo.

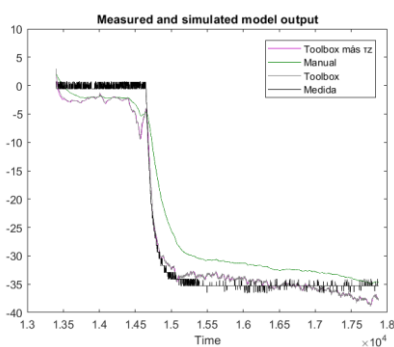


Ilustración 4.14 Ajuste para Calor bajo.

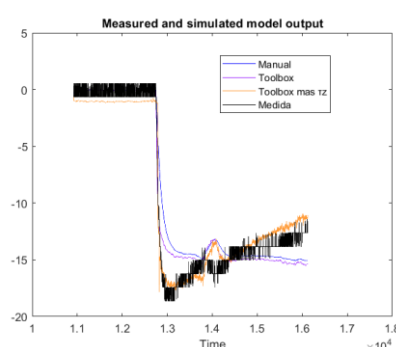


Ilustración 4.13 Ajuste para Calor medio.

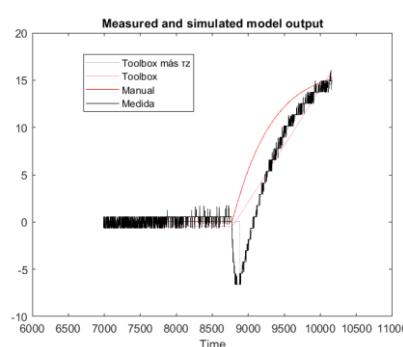


Ilustración 4.15 Ajuste para Calor alto.

En la Tabla 4.2 se recogen los valores de las constantes calculadas, estas constantes son las obtenidas de las funciones representadas en las ilustraciones de la parte superior, donde se incluyen el porcentaje de coincidencia con la medida de salida, la cual es denominada 'Best Fits'.

Tabla 4.2 Constantes obtenidas y porcentaje de conciencia de estas con señal de entrada  $T^a$  de aire de entrada.

		Placa Idle					
		Caudal mínimo					
		Calor bajo		Calor medio		Calor alto	
Manual	K	-9,34		-0,72		0,43	
	$\tau$	152,59		82,00		466,67	
	$\tau_d$	0,00		0,00		0,00	
	Best Fits (%)	72,83		73,44		45,07	
Toolbox	K	-10,11	-10,12	-0,73	0,00	4,78	0,45
	$\tau$	31,82	41,52	20,45	0,04	14021,00	465,40
	$\tau_d$	0,00	0,10	0,00	39,93	0,00	107,14
	$\tau_z$	0,00	9,12	0,00	-7312,60	0,00	-233,77
	Best Fits (%)	86,28	86,37	78,12	84,15	66,33	79,64

Observando la Tabla 4.2 se puede concluir, que se han realizado unos buenos ajustes, pero dichos ajustes son muy dependientes de las condiciones del problema, ya que se puede apreciar claramente que de unos a otros son modelos poco generales.

Estos mismos pasos se realizaron para una nueva señal de entrada tomándose ahora la medida del termopar que mide la temperatura del aire sobre el 'hot spot'. Puesto que el termopar que mide la temperatura del aire de entrada se encuentra fuera de la estructura, entonces, mientras la estructura está cerrada y no se introduce caudal, el termopar mide la temperatura ambiente de la habitación y

no está midiendo la temperatura interna de la estructura. Realizando este cambio de la señal de entrada, no se mejora el modelo, pero se puede probar que tal se adapta a la señal completa y no solo para el fragmento escogido para la realización de los cálculos (Ilustración 4.11). En la Ilustración 4.17 e Ilustración 4.16 se puede observar en concreto el ajuste para calor bajo, calculada de la misma forma que antes. Además, en el Anexo 0 se han recogido algunos de los resultados obtenidos.

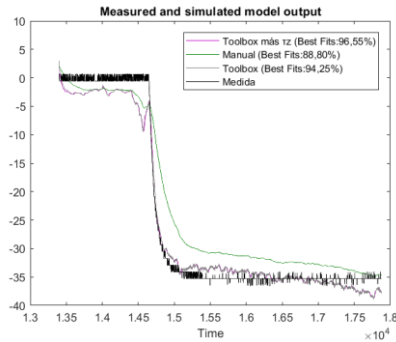


Ilustración 4.16 Ajuste para calor bajo.  
Entrada Tª aire sobre 'hot spot'.

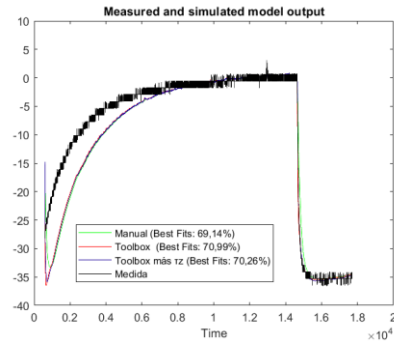


Ilustración 4.17 Ajuste señal total Calor bajo.  
Entrada Tª aire sobre 'hot spot'.

A pesar de obtener un buen ajuste mediante estas señales, igual que ocurría en el caso anterior se ha observado que los ajustes son muy dependientes de las condiciones del problema y no se pueden llegar a generalizar. Por lo tanto, se concluye que de esta forma no podemos llegar a obtener un modelo general que caracterice el sistema.

#### 4.4 Lecciones aprendidas

En primer lugar, se ha podido comprobar la importancia en la temperatura externa y, por lo tanto, el fundamento del trabajo. Además, se observado que refrigerar a altas temperaturas, obviamente depende de la temperatura del objeto que quieras enfriar, pero no siempre es necesario que sea un refrigerante frío ya que con el suficiente aporte de caudal de aire se es capaz de retirar una gran cantidad de calor.

Además, se han obtenido unos buenos resultados en la prueba de control con estrés de la placa, concluyendo que un control de temperatura PID regulado por la frecuencia es más eficaz y seguro que el control por defecto de dicha placa multiprocesador como los típicamente usados.

Los resultados obtenidos no han podido proporcionar una caracterización general, por lo que es posible que las consideraciones realizadas sean más complejas de lo que se supuso inicialmente. Cabe preguntarse entonces, que ya que hay múltiples efectos podríamos realizar un ajuste en base a otro tipo de señales de entrada, dividiendo el problema en varios subproblemas.

# 5 CONCLUSIONES

---

A lo largo de este proyecto se ha realizado un estudio térmico sobre la influencia de la temperatura exterior en una plataforma multiprocesador sin disipador. Desde el diseño de una plataforma de control térmico hasta el análisis de los datos obtenidos, pasando tanto por un trabajo de análisis previo como la elaboración de un sistema de adquisición de datos. Se puede afirmar que en este trabajo se han abordado objetivos muy importantes hoy en día para un ingeniero: investigación, diseño, puesta en marcha y análisis de resultados.

El trabajo ha combinado conocimientos de 3 áreas de trabajo como son la ingeniería automática, informática y termodinámica. Dentro de esta última área se ha realizado un esfuerzo en la comprensión y adaptación de los conocimientos adquiridos en un dispositivo *ciberfisico*. Por otro lado, la inclusión de otras áreas como la automática ha permitido evaluación e interpretación de los datos adquiridos. Por último, la inserción del área de la ingeniería informática ha posibilitado el estudio y comprensión sobre conocimientos del *kernel* de Linux, y la programación del código necesario para la adquisición y tratamiento de datos.

Se ha desarrollado una plataforma de control térmico de bajo coste acorde a los requisitos iniciales, que permite medir la temperatura de uno a 8 puntos de la estructura. Dicha estructura ha cumplido los objetivos planteados en su inicio, que son, la generación de una estructura de bajo coste aislada, lo suficiente como para poder someter la placa a un elevado estrés térmico, en un entorno de temperatura aproximadamente homogénea. El tratamiento de datos también ha sido una parte importante del trabajo, en el que se ha tenido que aprender el lenguaje de programación C, por otra parte, la plataforma de medición externa desarrollada se ha sincronizado con la medida de temperatura interna del procesador, pudiendo medir de forma precisa en varios puntos de la estructura.

En cuanto a los resultados obtenidos, no se ha podido llegar a obtener un modelo general para la perturbación generada, ya que los resultados obtenidos son muy dependientes de las condiciones externas. No obstante, los experimentos realizados nos han proporcionado información de interés en el comportamiento de este tipo de dispositivos, además, se comprobó que un control de temperatura PID regulado por la frecuencia es más eficaz y seguro que el instalado por defecto. Por ello, se deja para trabajo futuro, en primer lugar, volver a analizar la viabilidad de las simplificaciones



realizadas y, en segundo lugar, la obtención de dos modelos subdividiendo el problema en un modelo para la perturbación del caudal del aire y otro para la temperatura externa.

Además, podría usarse en el área de la ingeniería informática para que los alumnos puedan comprobar cómo le afectan las condiciones del entorno a una placa multiprocesador como la de un dispositivo *ciberfísico*.

## - BIBLIOGRAFÍA

- [1] Dieter Bohn. Apple confirms MacBook Pro thermal throttling, software fix coming today [En línea]. <<https://www.theverge.com/2018/7/24/17605652/macbook-pro-thermal-throttling-apple-software-fix>> [Consulta: 2019-05-16].
- [2] R. Bubbico, F. D'Annibale, B. Mazzarotta y C. Menale. «*Air Cooling of Li-ion Batteries: An Experimental Analysis*» En: Chemical Engineering transactions. Vol. 57, 2017. ISBN: 2283-9216. DOI: [10.3303/CET1757064](https://doi.org/10.3303/CET1757064).
- [3] R. Krishnarnurthy, A. Alvandpour, Borkar, S. «*High-performance and low-power challenges for sub-70 nm microprocessor circuits*». Custom Integrated Circuits Conference. Proceedings of the IEEE, 2002. ISBN: 0-7803-7250-6. Págs(s): 125 – 128. DOI: [10.1109/CICC.2002.1012781](https://doi.org/10.1109/CICC.2002.1012781).
- [4] Wang Q., Ping P., Zhao X., Chu G., Sun J., Chen C., 2012, Thermal runaway caused fire and explosion of lithium ion battery, Journal of Power Sources, 208, 210-224.
- [5] Xin Wang. *Intelligent Power Allocator*. Inf. téc. ARM, 2017.
- [6] Pablo Hernández Almudí. Diseño de un Governor basado en Control Inteligente de Temperatura. Proyecto de fin de carrera. Zaragoza, 2017.
- [7] Sergio Gutiérrez Verde. Aspectos térmicos de la ejecución de programas: estudio experimental sobre Pentium IV. Proyecto fin de carrera. Zaragoza, 2009.
- [8] Bubbico R., D Annibale F., Mazzarotta B., Menale C. «*A Simplified Model for improving Thermal Stability of Lithium-ion Batteries*» En: Chemicals Engineering Transactions., 2019. Volumen 77. Págs:817-822. DOI: [10.3303/CET1977137](https://doi.org/10.3303/CET1977137).
- [9] Valøen L.O., Reimers J.N. «*Transport properties of LiPF<sub>6</sub>-based Li-ion battery electrolytes*» En: Journal of Electrochemical Society, 2005. Vol. 152, Págs: 882-891.
- [10] Lou Y. *Nickel-metal hydride battery cooling system research for hybrid electric vehicle*. Shanghai Jiao Tong University. 2007.
- [11] Rami Sabbah, R. Kizilel, J.R. Selmán, S.Al-Hallaj. «*Active (air-cooled) vs. passive (phase change material) thermal management of high-power lithium-ion packs: limitation of temperature rise and*

*uniformity of temperature distribution*» En: Journal of Power Sources, 2008. Vol. 182, Issue 2. Págs: 630-638. DOI: [10.1016/j.jpowsour.2008.03.082](https://doi.org/10.1016/j.jpowsour.2008.03.082).

[12] Yang N., Zhang X., Li G., Hua D. «*Assessment of the forced air-cooling performance for cylindrical lithium-ion battery packs: a comparative analysis between aligned and staggered cell arrangements*» En: Applied Thermal Engineering, 2015. Vol. 80. Págs: 55-65. DOI: [10.1016/j.applthermaleng.2015.01.049](https://doi.org/10.1016/j.applthermaleng.2015.01.049).

[13] W. Yueh, Z. Wan, Y. Joshi y S. Mukhopadhyay. «*Experimental characterization of in-package microfluidic cooling on a System-on-Chip*». En: ISLPED. 2011 Págs. 43-48. DOI: [10.1109/ISLPED.2015.7273488](https://doi.org/10.1109/ISLPED.2015.7273488).

[14] Ling Z., Zhang Z., Shi G., Fang X., Wang L., Gao X., Fang Y., Xua T., Wang S., Liu X. «*Review on thermal management systems using phase change materials for electronic components, Li-ion batteries and photovoltaic modules*» En: Renewable and Sustainable Energy Reviews, Elsevier, Vol. 31(C), 2018. Págs: 427-438. 2014. DOI: [10.1016/j.rser.2013.12.017](https://doi.org/10.1016/j.rser.2013.12.017)

[15] Guo M., Sikha G., White R.E. «*Single-particle model for a lithium-ion cell: thermal behaviour*» En: Journal of Electrochemical Society, 158, 122-32. 2011. DOI: [10.1149/1.3521314](https://doi.org/10.1149/1.3521314).

[16] Bubbico R., D'Annibale F., Mazzarotta B., Menale C. «*Analysis of passive temperature control systems using PCM for application to secondary batteries cooling*» Journal of Thermal Science and Engineering, 2008. DOI: [10.1115/1.4040643](https://doi.org/10.1115/1.4040643).

[17] Park J., Lee S. y Cha H. «*App-Oriented Thermal Management of Mobile Devices*». En: ISLPED. 36:1-36:6. 2018. ISBN: 978-1-4503-5704-3. DOI: [10.1145/3218603.3218622](https://doi.org/10.1145/3218603.3218622).

[18] Pothukuchi R, Ansari A., Voulgaris P. y J. Torrellas. *Using Multiple Input, Multiple Output Formal Control to Maximize Resource Efficiency in Architectures*. En: ISCA. 2016. Págs. 658-670. 2016. DOI: [10.1109/ISCA.2016.63](https://doi.org/10.1109/ISCA.2016.63).

[19] M. Maggio, H. Hoffmann, M. D. Santambrogio, A. Agarwal y A. Leva. «*Controlling software applications via resource allocation within the heartbeats framework*». En: CDC. 2010. Págs. 3736-3741. DOI: [10.1109/CDC.2010.5717893](https://doi.org/10.1109/CDC.2010.5717893)

- [20] Silva E. Arquitecturas Alternativas para el Control de Procesos [Documento en línea]. <[http://profesores.elo.utfsm.cl/~esilva/Files/Arquitecturas\\_alternativas.pdf](http://profesores.elo.utfsm.cl/~esilva/Files/Arquitecturas_alternativas.pdf)> [Consulta: 2019-06-13].
- [21] T. Mudge. «Power: a first-class architectural design constraint». En: Computer 34.4 (2001), págs. 52-58. issn: 0018-9162. doi: 10.1109/2.917539.
- [22] Ángel Bernad Casalé. Estudio y modelo térmico de la refrigeración de microprocesadores Proyecto de fin de carrera. Zaragoza, 2012.
- [23] Weste N. y Harris D. CMOS VLSI Design: A Circuits and Systems Perspective. 4th. USA: Addison-Wesley Publishing Company, 2010. ISBN 13: 978-0-321-54774-3.
- [24] Ujaldón M. Arquitectura del PC. Universidad de Málaga, Vol. 5: Los chips. Cap. 31. 2003. ISBN: 85-95391-90-2.
- [25] Ujaldón M. Procesadores gráficos para PC, Capítulo 4, 2005. ISBN: 84-95391-09-0.
- [26] Martínez I. Convección térmica y másica. [Documento en línea]. <<http://webserver.dmt.upm.es/~isidoro/bk3/c12/Conveccion%20termica%20y%20masica.pdf>> [Consulta: 2019-04-13].
- [27] GitHub: lm-sensors/lm sensors [En línea]. <<https://github.com/lm-sensors/lm-sensors>> [Consulta: 2019-01-10].
- [28] Arafet P. y Chang F. Métodos de identificación dinámica. Facultad de Ingeniería Eléctrica. Universidad de Oriente. 2008.
- [29] Alcampo: Espuma de poliuretano 750 mililitros HAUSER [En línea] <<https://www.alcampo.es/compra-online/hogar/bricolaje-e-iluminacion/adhesivos-y-siliconas/reparadores/hauser-espuma-de-poliuretano-750-mililitros/p/590854>> [Consulta: 2019-04-02].
- [30] Pico Technology-USB TC-08 Thermocouple Data Logger [En línea]. <<https://www.picotech.com/data-logger/tc-08/thermocouple-data-logger>> [Consulta: 2019-03-10].
- [31] Flores A. Obtención de la Función de Transferencia de Sistemas mediante la Identificación Paramétrica a partir de datos Experimentales, Instituto tecnológico de Mérida, 2013.

[32] George Stephanopoulos. Chemical Process Control: An Introduction to Theory and Practice. 3<sup>rd</sup> Edition, 1985. Vol. 21, Issue 4. Págs:502-504. DOI: [10.1016/0005-1098\(85\)90090-1](https://doi.org/10.1016/0005-1098(85)90090-1).

[33] Simulación de Procesos Udenar: Control proporcional [En línea].

[<https://simulaciondeprocesosudenar.wordpress.com/control-proporcional-integral-derivativo-pid-y-estabilidad-de-sistemas-dinamicos/>](https://simulaciondeprocesosudenar.wordpress.com/control-proporcional-integral-derivativo-pid-y-estabilidad-de-sistemas-dinamicos/) [Consulta: 2019-04-13].

[34] Pico Technology-USB TC-08 logger PicoLog 6 software [En línea].

[\[https://www.picotech.com/data-logger/tc-08/usb-tc-08-logger-picolog-6-software\]](https://www.picotech.com/data-logger/tc-08/usb-tc-08-logger-picolog-6-software) [Consulta: 2019-03-11].

[35] MathWorks: textscan [En línea]. [\[https://es.mathworks.com/help/matlab/ref/textscan.html\]](https://es.mathworks.com/help/matlab/ref/textscan.html)

[Consulta: 2019-04-22].

[36] Inteltec: Cámara Climática de Sobremesa proporcional [En línea]

[\[http://www.ineltec.es/es/productos/equipos-de-simulacion/camaras-climaticas/camara-climatica-sobremesa#\]](http://www.ineltec.es/es/productos/equipos-de-simulacion/camaras-climaticas/camara-climatica-sobremesa#) [Consulta: 2019-02-28].

# A. PLATAFORMA Y OBTENCIÓN DE DATOS

## A.1 Evolución de la estructura

Se optó en primera instancia por una caja de zapatos en la que se introdujo la plataforma multiprocesador. A continuación, se decidió por donde se iban a introducir tanto los cables de esta como los termopares. Por otro lado, la localización y fijación de la plataforma es importante para que las pruebas sean reproducibles.

Con el objetivo de que la influencia de la presión sea prácticamente despreciable, el aire que entre lo hará a una distancia de 10 cm y realice la función de entrada y salida del aire. Se pretendió analizar el funcionamiento de la plataforma con la caja abierta. Para ello, en primer lugar, se ejecutaron pruebas como la que se muestra en la Ilustración A.1 con carga de trabajo.

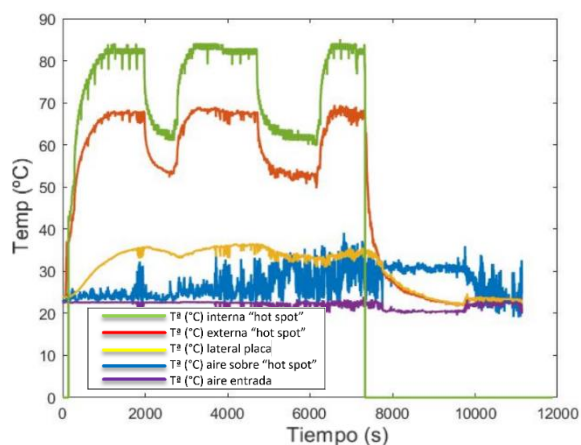


Ilustración A.1 Estructura abierta. Con carga.  
Caudal nulo.

Se enciende la placa con la caja abierta; se comienza a leer temperaturas; se enciende la carga de trabajo hasta que acaban todas las iteraciones; sucesivamente se realiza dos veces más y se apaga la placa; cómo se puede apreciar en 5 minutos la temperatura del 'hot spot' desciende 30°C al apagarse la placa, después de haber puesto en marcha la carga de trabajo; esto es debido al gradiente de temperatura de la placa con el aire, por lo que la temperatura desciende rápido.

Por otra parte, se ejecutaron pruebas sin la placa multiprocesador en el que poder comprobar la homogeneidad del aire con la estructura cerrada. Los resultados se muestran en la Ilustración A.3.

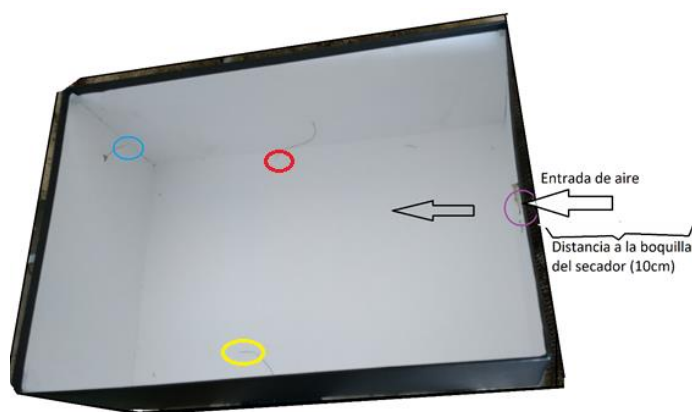


Ilustración A.2 Caja antes de ser aislada.

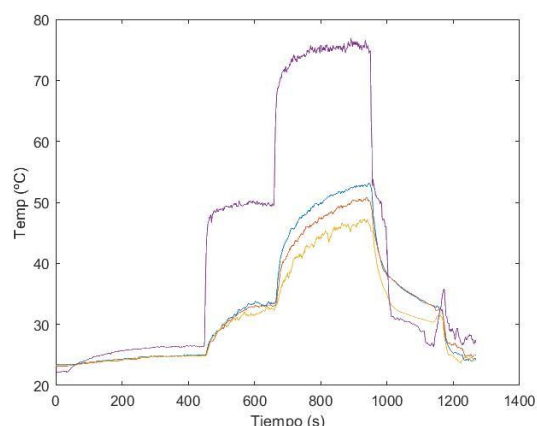


Ilustración A.3 Estructura cerrada. Placa apagada.

En la Ilustración A.3 se pueden observar para las tres posiciones de calor del secador, cómo se comporta el sistema y cuál es su capacidad de retener el calor con la estructura cerrada. En definitiva, se concluye, en primer lugar, que no es lo suficientemente homogénea como para poder considerar una temperatura constante en el interior y, en segundo lugar, el calor se trasfiere al exterior rápidamente, ya que en 30 segundos la temperatura desciende una media de 12 °C. Esto lo podemos comprobar también en la Ilustración A.4.

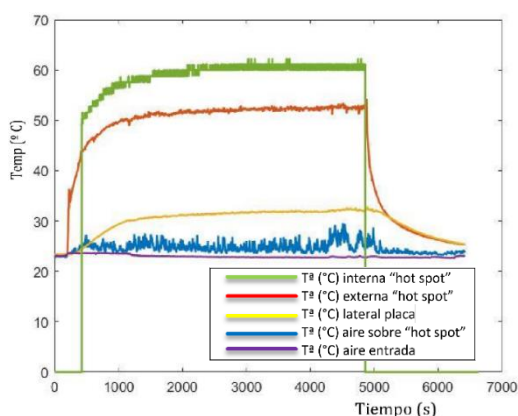


Ilustración A.4 Estructura cerrada. Caudal nulo.

Por lo tanto, se decidió aislar la estructura a partir de un encofrado de espuma de poliuretano. En la Ilustración Ilustración A.5 se ha realizado una prueba en la que se pretende comparar la mejora de esta.

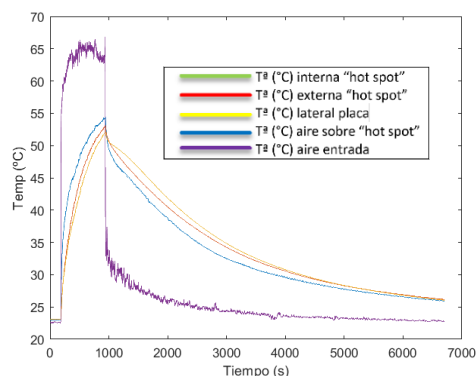


Ilustración A.5 Estructura cerrada aislada. Placa apagada

Ahora, en los primeros 15 minutos se puede observar que la temperatura ha descendido  $11^{\circ}\text{C}$  y en los siguientes 15 min  $6,5^{\circ}\text{C}$ . Además, la homogeneidad en las medidas es mucho mayor. Por lo que se ha comprobado la notable mejora gracias al encofrado realizado. De la misma manera se puede observar para las tres posiciones de caudal y la placa encendida, en la que con *Calor alto* se acaba apagando para no dañarse.

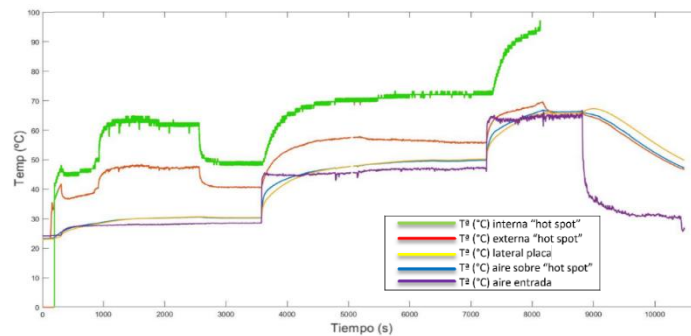


Ilustración A.6 Placa encendida. Calor bajo, Calor medio y Calor alto. Estructura cerrada.

## A.2 Sistema de medición del sensor del procesador

En cuanto a la obtención de la temperatura del procesador, esta se realiza a partir de la librería *Im-sensors*. Estas bibliotecas permiten obtener muchas de las características del sistema recogidas por los sensores de la placa base y del procesador, dependiendo del modelo de plataforma.

Una vez que se han encontrado el modelo del procesador y los campos del sensor de este, se aprovecha toda la API ('Application Programming Interface') de *Im-sensors*, accediendo al campo concreto que proporcione la temperatura del procesador, con el que podamos trabajar con él fácilmente. Además, se recoge también al mismo tiempo la frecuencia de trabajo del procesador, para poder comprobar a que frecuencia se trabaja en todo momento.

Con el fin de poder obtener los datos, se ejecuta las instrucciones de un código, dicho código para cada tiempo de muestreo vuelca ambos datos en un fichero de texto.

En el trabajo de Gutierrez S. [7] se comprobó, que las interferencias que se generan en el procesador son mínimas y, además, la ejecución no se ve afectada por el hilo que recoge los datos.

Por otro lado, existen dos sensores en el procesador, uno dentro del propio procesador, y otro en el empaquetado (socket) del procesador. Pero se comprobó que ambas medidas daban aproximadamente la misma temperatura.



## A.3 Código desarrollado para la obtención de los datos del procesador

### A.3.1 Ejecutable fijar frecuencia

```
#!/bin/bash

echo userspace >/sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo userspace >/sys/devices/system/cpu/cpu1/cpufreq/scaling_governor
echo userspace >/sys/devices/system/cpu/cpu2/cpufreq/scaling_governor
echo userspace >/sys/devices/system/cpu/cpu3/cpufreq/scaling_governor

# 1 para activar CPU 0 para desactivar
# De normal cpu0 y cpu1 con 1
# Despues de echo poner 0 o 1
# El del cpu0 no tocar

echo 1 >/sys/devices/system/cpu/cpu0/online
echo 1 >/sys/devices/system/cpu/cpu1/online
echo 0 >/sys/devices/system/cpu/cpu2/online
echo 0 >/sys/devices/system/cpu/cpu3/online

# Despues de echo poner la frecuencia

echo 810000 >/sys/devices/system/cpu/cpu0/cpufreq/scaling_setspeed
echo 810000 >/sys/devices/system/cpu/cpu1/cpufreq/scaling_setspeed
echo 810000 >/sys/devices/system/cpu/cpu2/cpufreq/scaling_setspeed
echo 810000 >/sys/devices/system/cpu/cpu3/cpufreq/scaling_setspeed

echo "Todo bien"

sleep 10
```

### A.3.2 Ejecutable leer temperatura y frecuencia

```
#!/bin/bash

# Check if we are root

[ "$(whoami)" != "root" ] && exec sudo -- "$0" "$@"

# Default filename

filename="temps_log.txt"

# Change file

if [ "$#" -gt 0 ];
```

```
then
    filename=$1
fi
while true;
do
    cat/sys/class/thermal/thermal_zone0/temp/sys/devices/system/cpu/cpu0/cpufreq
/cpufreq_info | tr "\n" " " >>$filename
sleep 1
done
```

### A.3.3 Ejecutable PID

```
#!/bin/bash

echo 0 >/sys/devices/system/cpu/cpu2/online
echo 0 >/sys/devices/system/cpu/cpu3/online

echo userspace >/sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo userspace >/sys/devices/system/cpu/cpu1/cpufreq/scaling_governor
/home/linaro/governor-tfmg/PID_controller/user_governor

echo ondemand >/sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo ondemand >/sys/devices/system/cpu/cpu1/cpufreq/scaling_governor

echo 1 >/sys/devices/system/cpu/cpu2/online
echo 1 >/sys/devices/system/cpu/cpu3/online
```

# B. CONTROLADOR PID

## B.1 Concepto y funcionamiento de un controlador PID

En este apartado se explica brevemente cuál es el funcionamiento del control PID y el control que se implementó en la placa sustituyendo al control por defecto que disponen los procesadores.

Un controlador es aquel que regula una magnitud física, esta magnitud depende del caso que se estudia. Su complejidad es función de los parámetros que influyen en el sistema. En concreto, un control PID, viene determinado por tres parámetros. En primer lugar, el término proporcional (*P*), el cual actúa corrigiendo de forma proporcional al error. En segundo lugar, el término integral (*I*), que corrige el error de forma proporcional a la integral del error, permite reducir el error a cero, si se aplica un esfuerzo de control suficiente. En tercer lugar, el término derivativo (*D*), genera una acción

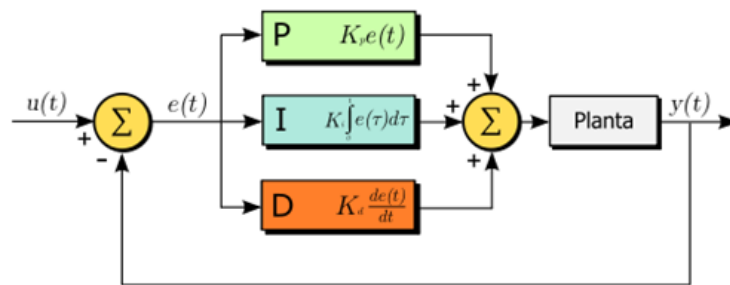


Ilustración B.1 Esquema funcionamiento control PID. [33]

de control sobre el cambio de rango de control de forma proporcional, este término en sistemas con ruido de fondo no es útil, ya que desestabiliza el sistema de forma que el elemento final de control sufre y reduce su vida útil. Este control puede describirse mediante el siguiente esquema:

En la placa multiprocesador con la que se trabaja en este proyecto se ha implementado un control PID, el cual mediante la modificación de la frecuencia se controla la temperatura del procesador, esto fue realizado por Pablo Hernández [6] en su trabajo de fin de grado.

La función de transferencia que se considera, en la que no se tiene en cuenta la temperatura externa, es la siguiente:

$$G(s) = \frac{T_{\Delta P}(s)}{f(s)} = \frac{K}{1 + \tau \cdot s} e^{-\tau_d \cdot s} \quad [\text{Ecuación B.1}]$$

Con este sistema, se obtuvo los parámetros de control PID que se muestran en la Ilustración B.1;  $K_p=100k$ ,  $K_i=0.00555$  y  $K_d=0$ . Con los cuales en el apartado *Control con estrés de la placa* se realizará una prueba.

# C. CÓDIGO Y RESULTADOS EN MATLAB

---

## C.1 Código en Matlab

### C.1.1 Tratamiento temperatura-frecuencia

```
fileID = fopen('2-111(2).txt');
formatSpec = '%d';
B = fscanf(fileID, formatSpec); % Aqui se lee el fichero
fclose(fileID);

Ts = 1.0; %Colocar aqui la frecuencia de muestro

Temp=B(1:2:end,:);
Freq=B(2:2:end,:);

Temp = rdivide(Temp,1000); %Para pasar a grados

data = iddata(Temp,Freq,Ts) %pt se?al de salida (temp) %ct se?al de
entrada (freq) %Ts intervalo entre medidas
data.InputName = '\Delta Freq';
data.InputUnit = 'KHz';
data.OutputName = '\Delta PTemp';
data.OutputUnit = 'C';
data.TimeUnit = 'seconds';

set(0,'defaultlinelinenewidth',1.5)

figure(1);
opt = iddataPlotOptions('time');
opt.TimeUnit = 'seconds';
opt.Grid = 'on';
h = plot(data,opt,'-')
get(h)
xlabel('Time');
title('');
```

### C.1.2 Tratamiento temperatura interna procesador

El fichero de texto mediante la función *fscanf*. Y consecutivamente, mediante la función *iddata* que nos permite encapsular valores de entrada y salida del sistema. En nuestro caso se registran los valores de frecuencia y temperatura con una frecuencia de muestreo de 1 segundo.

```

fileID = fopen('2-111(2).txt');
formatSpec = '%d';
B = fscanf(fileID, formatSpec); % Aquí se lee el fichero
fclose(fileID);

Ts = 1.0; %Colocar aquí la frecuencia de muestro

Temp=B(1:2:end,:);
Freq=B(2:2:end,:);

Temp = rdivide(Temp,1000); %Para pasar a grados Celsius

data = iddata(Temp,Freq,Ts) %Se almacenan: Señal de salida Temperatura
(Temp) %Señal de entrada de Frecuencia (freq) %Frecuencia de muestreo
Ts
data.InputName = '\Delta Freq';
data.InputUnit = 'KHz';
data.OutputName = '\Delta PTemp';
data.OutputUnit = 'C';
data.TimeUnit = 'seconds';

set(0,'defaultlinelength',1.5)

figure(1);
opt = iddataPlotOptions('time');
opt.TimeUnits = 'seconds';
opt.Grid = 'on';
h = plot(data,opt,'-')
get(h)
xlabel('Time');
title('');

```

### C.1.3 Tratamiento temperaturas externas

Mediante la función *textscan* [35] lee los datos del archivo de texto en un array de celdas, separando los datos de los caracteres no numéricos. Luego, se crea la matriz de ceros en la que se introducen cada temperatura.

```

fildID = fopen('8real.csv');
a = fgetl(fildID);
datos=textscan(fildID,"""%8c""","""%6c""","""%6c""","""%6c""","""%6c""","""%6c""","""%
6c""","""%6c""","""%6c""");
%Cambiar el número de ""%6c"" en el caso de cambiar el número de
termopares usados, tantos como termopares usados
numDatos = size(datos{2},1); %Se define el tamaño de la matriz de
temperaturas
Vtemp = zeros(numDatos,8); %Cambiar según el número de termopares usados
j=1;
w=2;
%Bucle de asignación de temperaturas en la matriz:
while j<=8 %Poner el número de termopares usados
for i=1:numDatos
    num1 = strrep(datos{w}(i,:),',','.');
    Vtemp(i,j) = str2num(num1);
end
j=1+j;

```

```
w=1+w;
end
```

A continuación, se crea la matriz de tiempo en el que se introducen en segundos los respectivos tiempos de cada medida de temperatura. Y finalmente, la posterior representación con la que se comprueba el funcionamiento.

```
numTiempo = size(datos{2},1); %Se define el tamaño de la matriz del tiempo
Ts = zeros(numTiempo,1);
t=1;
%Bucle de asignación del tiempo en la matriz:
for y=1:numTiempo
    Ts(y,1)=3600*str2num(datos{1}(y,1:2))+60*str2num(datos{1}(y,4:5))+str2num(datos{1}(y,7:8));
    t=1+t;
end

p=plot(Ts,Vtemp); %Representación Temperatura vs Tiempo

fclose(fildID);
```

#### C.1.4 Tratamiento sincronización de medidas

En el siguiente código se muestran los fragmentos del código del tratamiento de los datos internos con las modificaciones realizadas:

```
%TRATAMIENTO DE DATOS INTERNOS
fileID = fopen('2-111.txt');
formatSpec = '%d';
B = fscanf(fileID, formatSpec); %Aquí se lee el fichero
fclose(fileID);
Temp=B(1:2:end,:);
Freq=B(2:2:end,:);
Temp = rdivide(Temp,1000);
Tsx = 1.035; %Colocar aquí la frecuencia de muestro real media
data = iddata(Temp,Freq,Tsx) %Se almacenan: Señal de salida Temperatura
(Temp) %Señal de entrada de Frecuencia (freq) %Frecuencia de muestreo Ts
```

Ahora,  $T_{sx}$  es la frecuencia de muestreo real media, esta se halla para cada prueba. Habrá que calcularla con las mediciones descritas anteriormente. En el siguiente fragmento se crea una matriz,  $T_{ss}$ , dicha longitud debe ser la misma que la obtenida con los datos de temperatura externa, ya que para poder representar los datos en *Matlab* es imprescindible.

Posteriormente, se realiza la corrección del tiempo de retraso hasta que se comienza a leer temperaturas, mediante el sensor interno. En este caso esta corrección se le ha denominado *offset*, el cual habrá que introducirlo en segundos. De la misma forma que antes con  $T_{ss}$ , la matriz de temperatura interna debe tener exactamente la misma longitud que la matriz de temperaturas

externas, esto se realiza haciendo unos pequeños cálculos, como se puede ver en el código de la parte inferior, mediante la variable  $m$ , ya que  $m$  no puede ser una matriz y es el que marca cuando se deja de rellenar esta.

```
numTime = size(datos{2},1); %Se define el tamaño de la matriz del tiempo
Tss = zeros(numTime,1);
t=0;
%Bucle en el que se rellena la matriz de tiempo interna con Tsx
for g=1:numTime
    Tss(g,1) = t;
    t=Tsx+t; %%*****
end

%CORRECCIÓN DEL ERROR OFFSET
Vtemp=Vtemp; %Se guarda en otra variable la temperatura externa
h = 0;
m = size(data.OutputData);
offset=340; %Colocar tiempo de retraso hasta que se comienza a leer
op=(offset+1);
ope=op/Tsx;
ope=round(ope);
opem=ope-1;
m=[opem,0]+m;
for s=ope:m
    h=h+1;
    Vtemp(s,j) = data.OutputData(h);
end
```

Una vez hemos resuelto el problema del error constante, se realiza la interpolación de los datos de temperatura interna respecto a la matriz  $Tss$ , que corresponde a la frecuencia real de muestreo. Dicha interpolación se realiza mediante la función *interp*, como vemos en el siguiente fragmento de código. Por último, se realiza la representación final.

```
%INTERPOLACIÓN
l=size(Vtemp(:,5))-1; %Número de temperaturas a interpolar
Vinterp=[0:1:l]';
VTinterp5=interp1(Tss,Vtemp(:,5),Vinterp); %Se interpolan las
temperaturas para la frecuencia de muestreo Tss

p=plot( Ts, Vtemp, Tss, Vtemp(:,5)); %REPRESENTACIÓN FINAL

%CONDICIONES NULAS PARA CADA PRUEBA
VTinterp5o=VTinterp5-79.7;
Vtempo=Vtemp(:,4)-22.318;

Vtempo2=Vtemp(:,3)-46.78;
Vtempo3=Vtemp(:,1)-38.988;

ylabel('Temp (°C)');
xlabel('Tiempo (s)');
fclose(fildID);
```



## C.2 Pruebas realizadas para la obtención de los parámetros del sistema

### • ESTRUCTURA ABIERTA

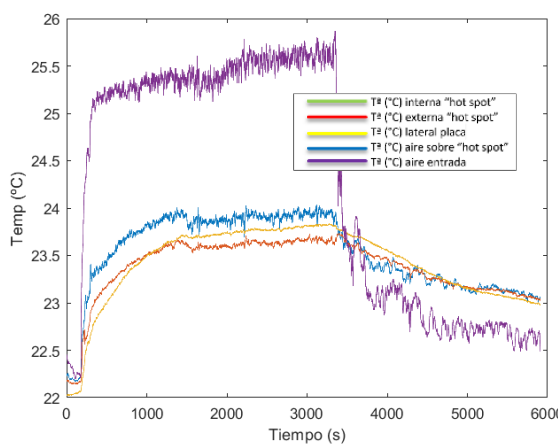


Ilustración C.1 Calor medio. Placa apagada.

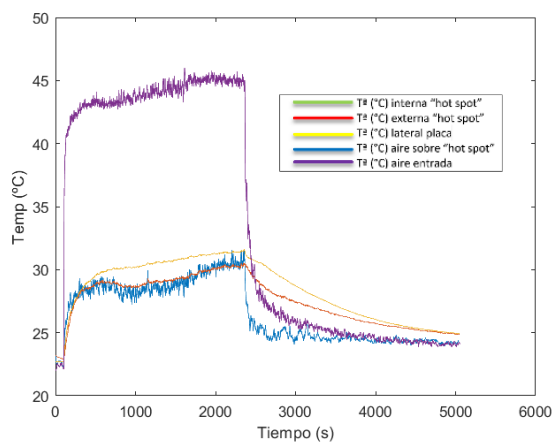


Ilustración C.2 Calor medio. Placa apagada.

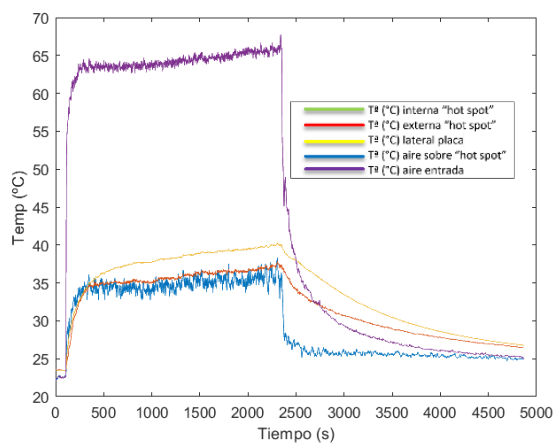


Ilustración C.3 Calor alto. Placa apagada.

## a. Caudal mínimo.

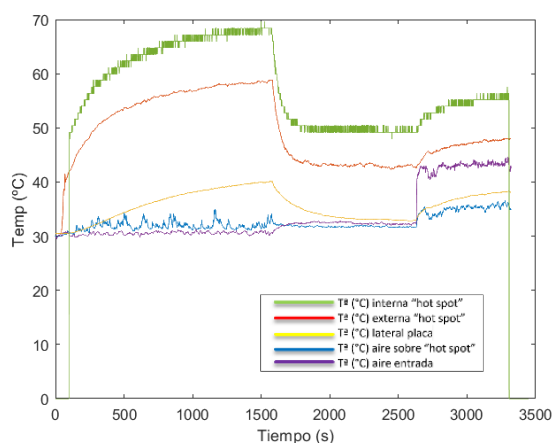


Ilustración C.4 Calor bajo y medio.

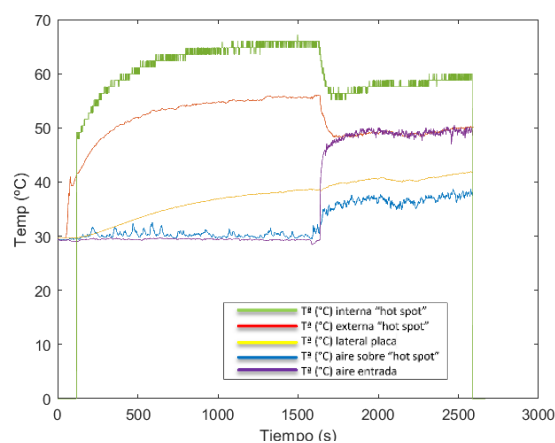


Ilustración C.5 Calor medio.

## b. Caudal máximo

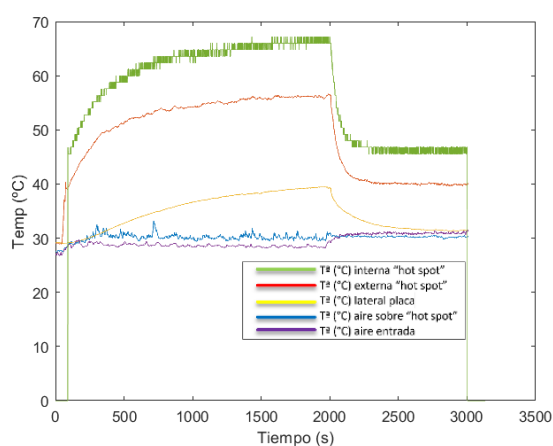


Ilustración C.6 Calor bajo.

## ● ESTRUCTURA CERRADA

## a. Caudal mínimo.

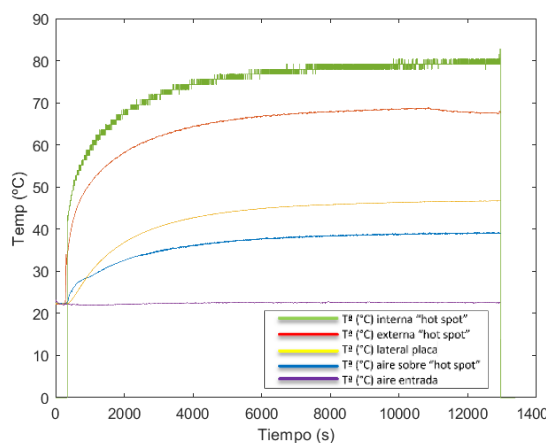


Ilustración C.7 Caudal nulo. Hasta Est. Estacionari

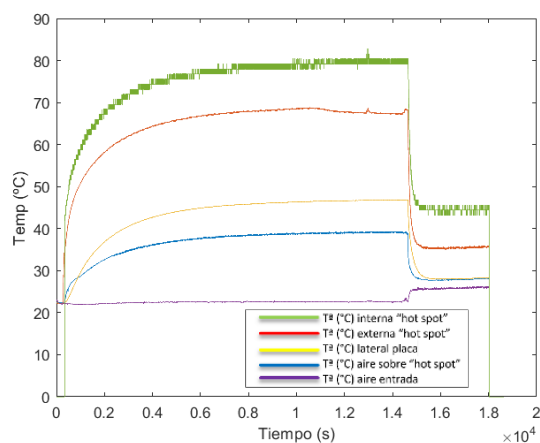
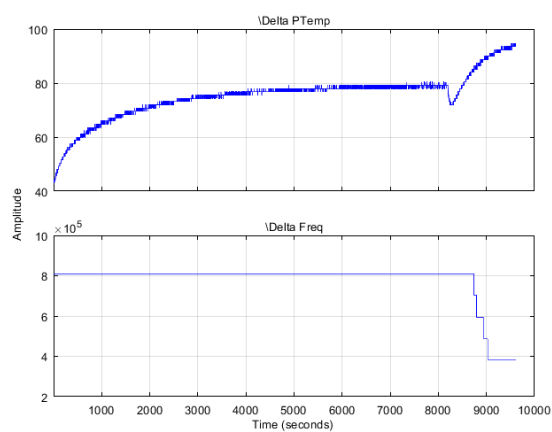
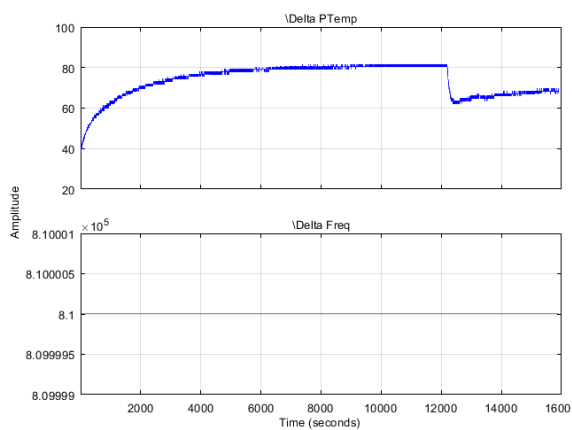
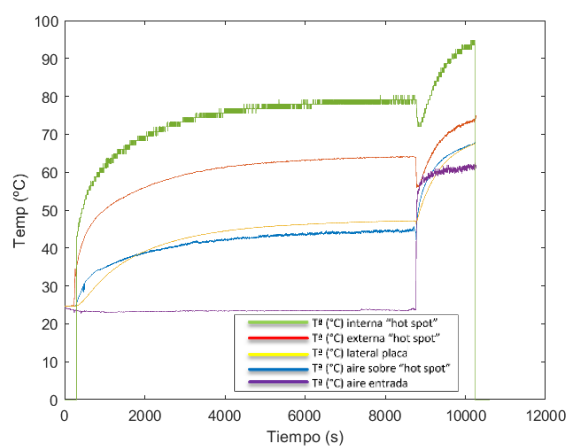
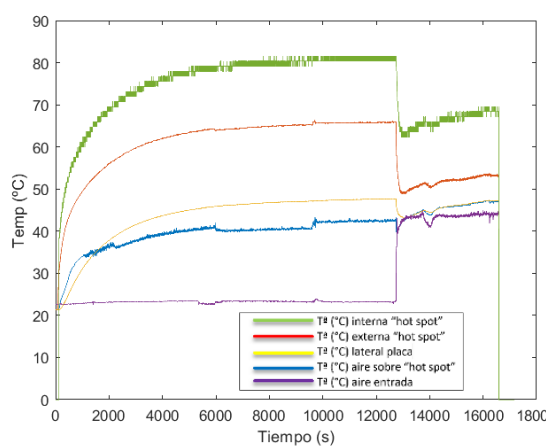
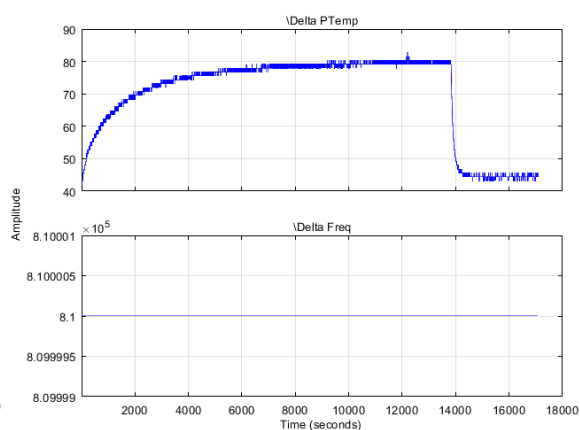
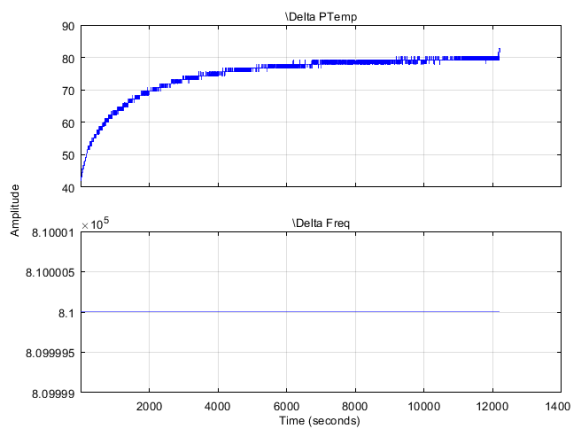


Ilustración C.8 Calor bajo.



## b. Caudal máximo.

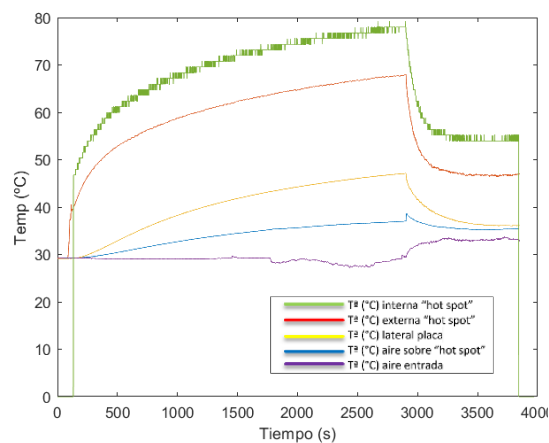


Ilustración C.15 Calor bajo.

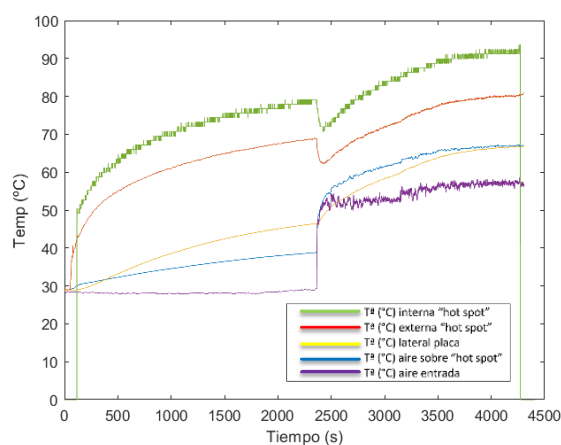


Ilustración C.16 Calor medio.

## • ESTRUCTURA CERRADA Y MULTIPLICADOR

## a. Caudal mínimo

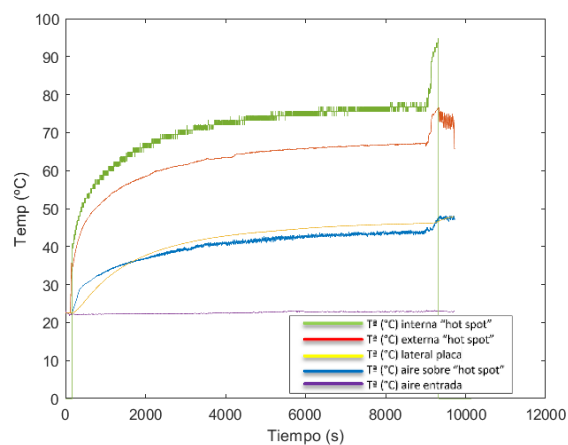


Ilustración C.17 Caudal nulo.

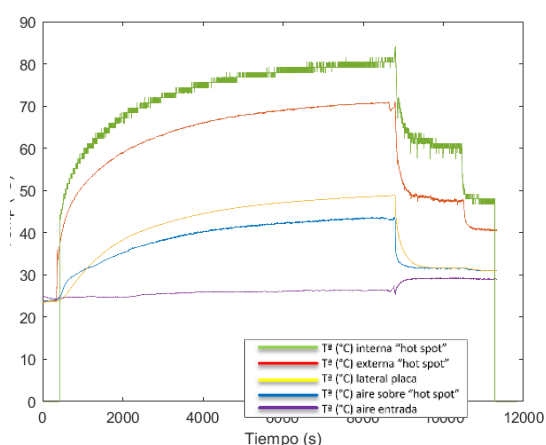


Ilustración C.18 Calor bajo.

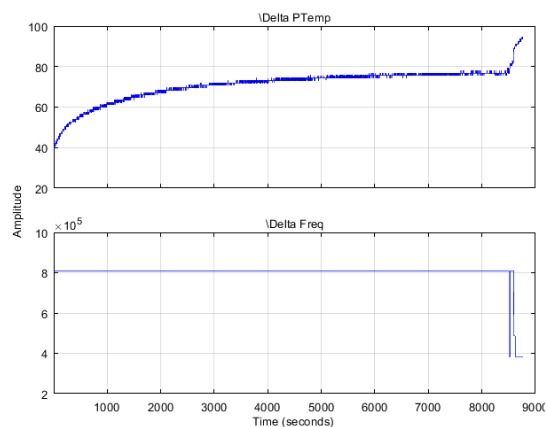


Ilustración C.19 Calor nulo. Frecuencia fija.

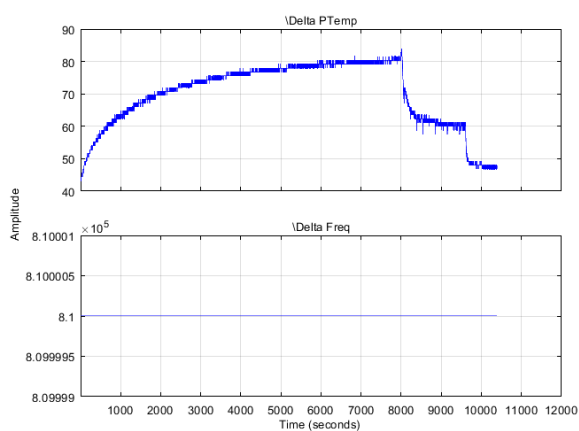


Ilustración C.20 Calor bajo. Frecuencia fija.

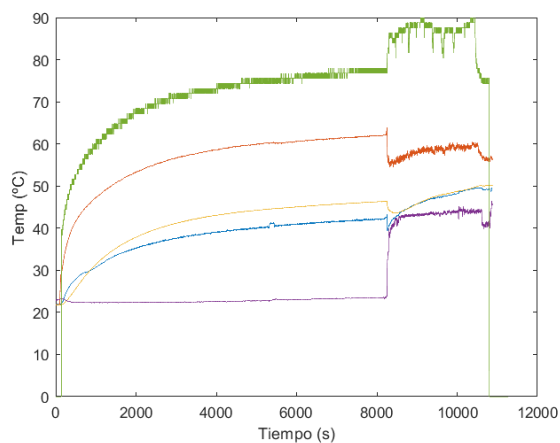


Ilustración C.21 Calor medio.

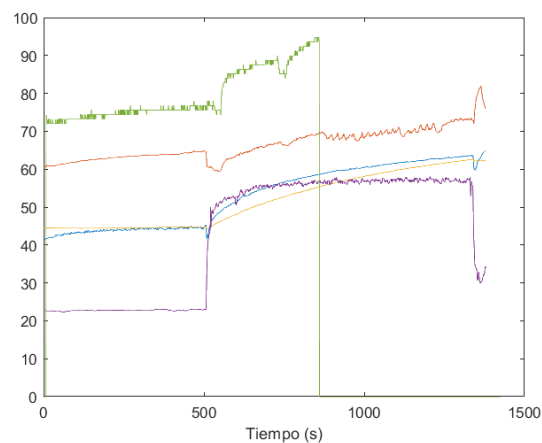


Ilustración C.22 Calor alto.

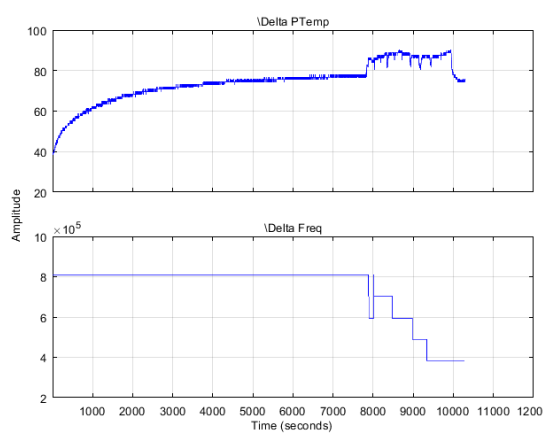


Ilustración C.23 Calor medio. Frecuencia fija.

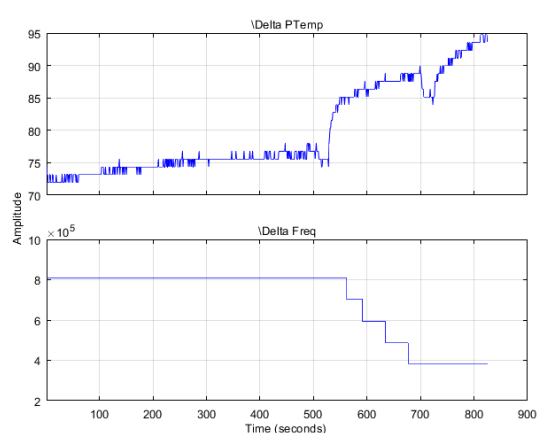


Ilustración C.24 Calor alto. Frecuencia fija.

### C.3 Pruebas para la comparación entre el control por defecto y el PID

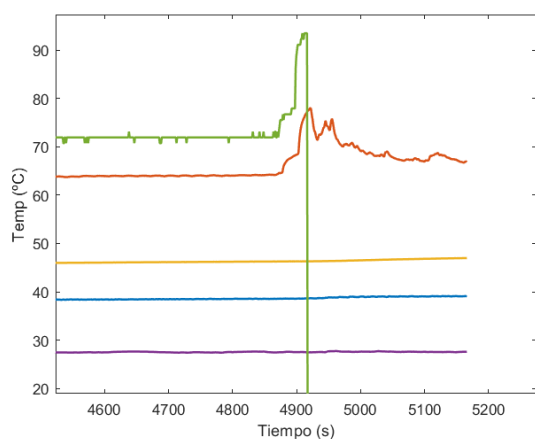


Ilustración C.25 Por defecto.

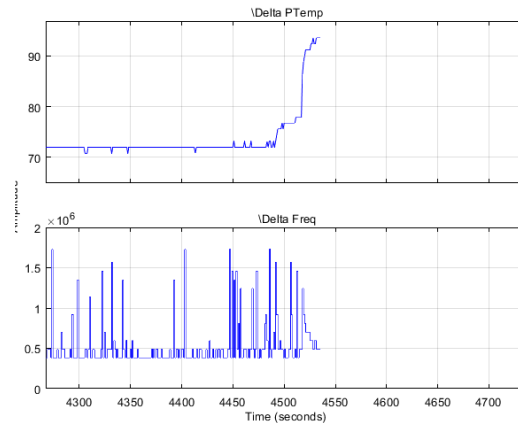


Ilustración C.26 Por defecto. Frecuencia no fija.

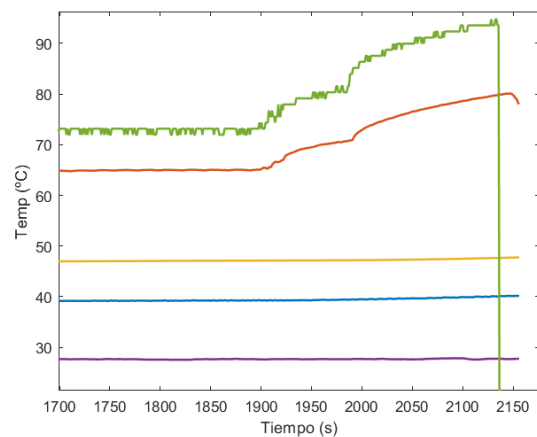


Ilustración C.27 Con PID. Temperatura objetivo 85 °C.

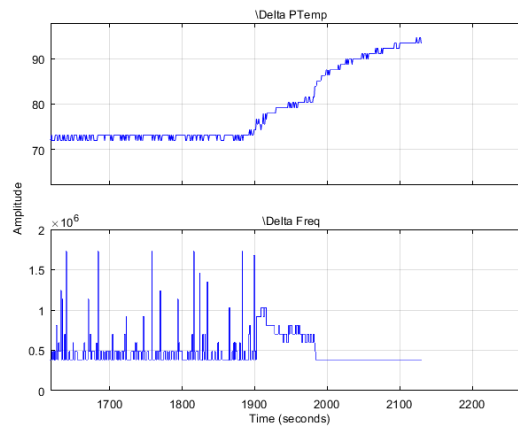


Ilustración C.28 Con PID. Frecuencia fijada. Temperatura objetivo 85 °C.

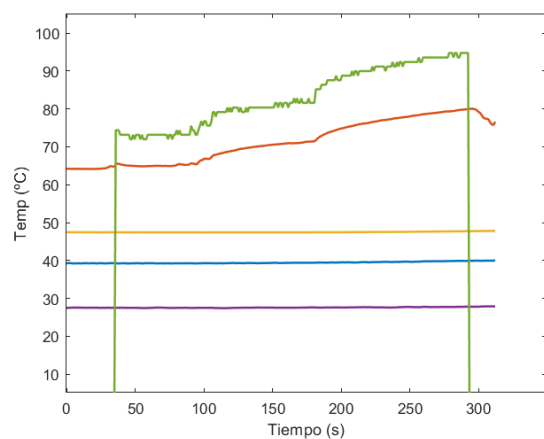


Ilustración C.29 Ilustración C.28 Con PID. Temperatura objetivo 70 °C.

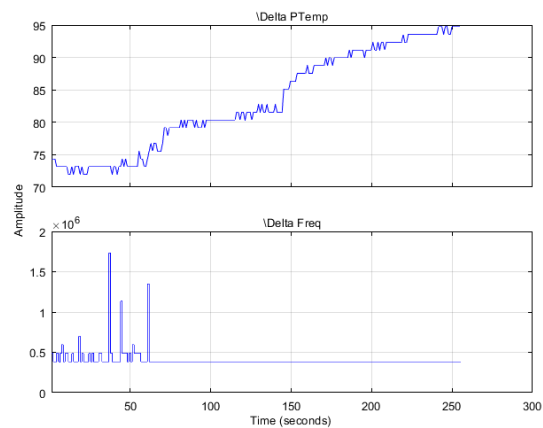


Ilustración C.29 Ilustración C.28 Con PID. Frecuencia fijada. Temperatura objetivo 70 °C.

## C.4 Gráficas de ajustes obtenidos

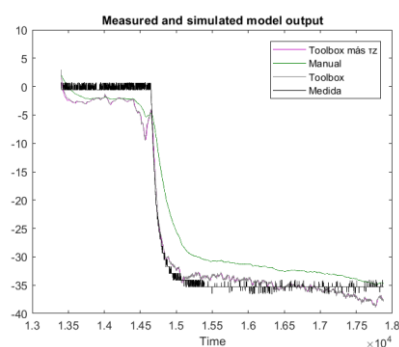


Ilustración C.30 Ajuste señal Calor bajo. Entrada Tª aire.

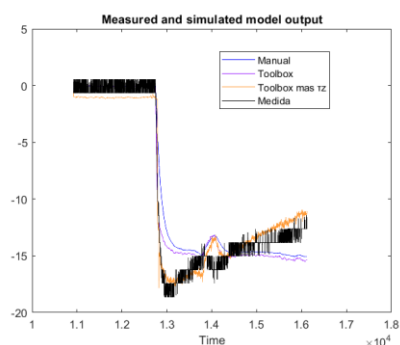


Ilustración C.31 Ajuste señal Calor medio. Entrada Tª aire.

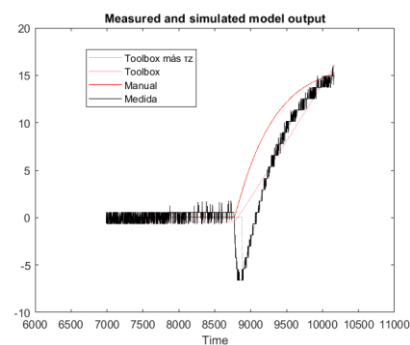


Ilustración C.32 Ajuste señal Calor alto. Entrada Tª aire.

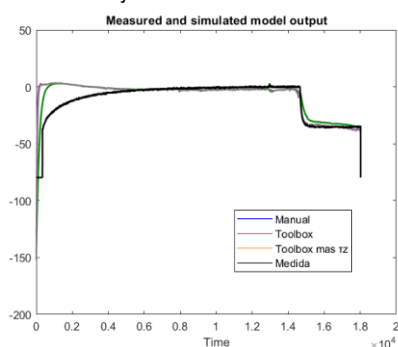


Ilustración C.33 Ajuste señal total Calor bajo. Entrada Tª aire.

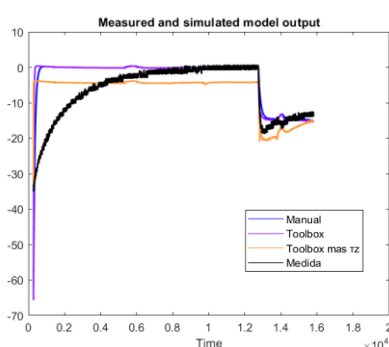


Ilustración C.34 Ajuste señal total Calor medio. Entrada Tª aire.

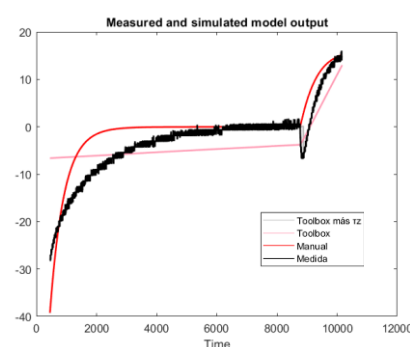


Ilustración C.35 Ajuste señal total Calor alto. Entrada Tª aire.

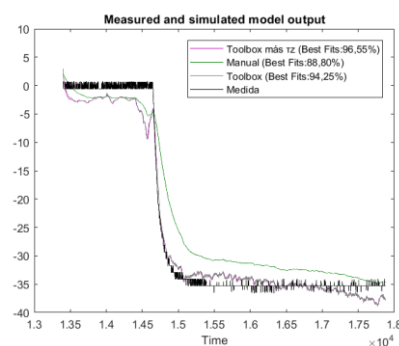


Ilustración C.36 Ajuste señal Calor bajo. Entrada Tª aire sobre 'hot spot'.

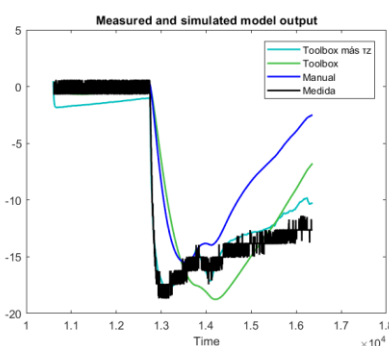


Ilustración C.37 Ajuste señal Calor medio. Entrada Tª aire sobre 'hot spot'.

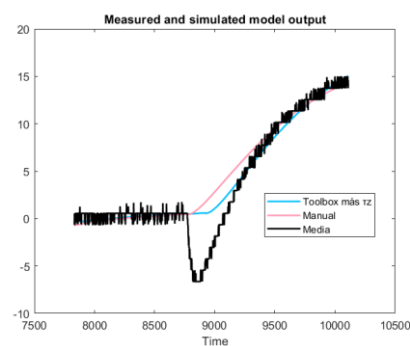


Ilustración C.38 Ajuste señal Calor alto. Entrada Tª aire sobre 'hot spot'.

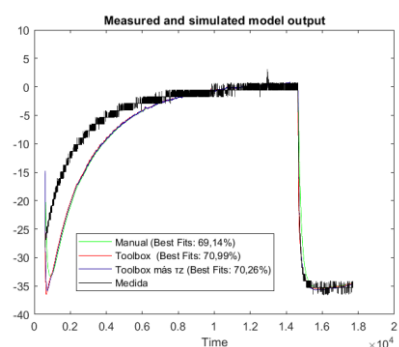


Ilustración C.39 Ajuste señal total Calor bajo. Entrada Tª aire sobre 'hot spot'.

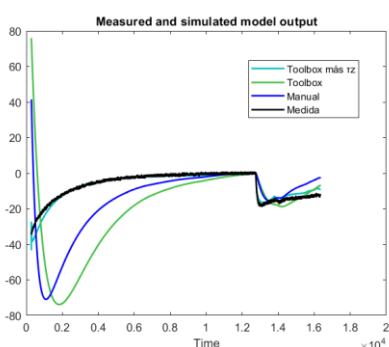


Ilustración C.40 Ajuste señal total Calor medio. Entrada Tª aire sobre 'hot spot'.

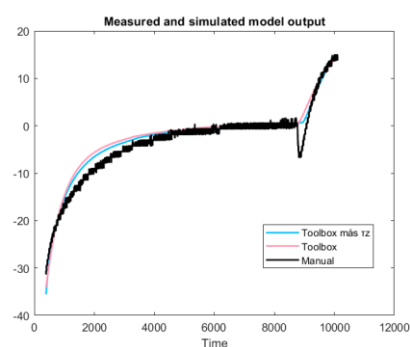


Ilustración C.41 Ajuste señal total Calor alto. Entrada Tª aire sobre 'hot spot'.